

Cosminexus アプリケーションサーバ V8

リファレンス コマンド編

文法書

3020-3-U14-21

マニュアルの購入方法

このマニュアル，および関連するマニュアルをご購入の際は，
巻末の「ソフトウェアマニュアルのサービス ご案内」をご参
照ください。

対象製品

適用 OS : Windows Server 2003 , Windows Server 2003 R2 , Windows Server 2003 (x64) , Windows Server 2003 R2 (x64) , Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2

P-2443-7D84 uCosminexus Application Server Standard 08-50

P-2443-7K84 uCosminexus Application Server Enterprise 08-50

P-2443-7M84 uCosminexus Web Redirector 08-50

P-2443-7S84 uCosminexus Service Platform 08-50

適用 OS : Windows Server 2003 , Windows Server 2003 R2 , Windows Vista , Windows XP , Windows 7

P-2443-7E84 uCosminexus Developer Standard 08-50

P-2443-7F84 uCosminexus Developer Professional 08-50

P-2443-7T84 uCosminexus Service Architect 08-50

適用 OS : Windows Server 2003 , Windows Server 2003 R2 , Windows Server 2003 (x64) , Windows Server 2003 R2 (x64) , Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows Vista , Windows XP , Windows 7

P-2443-7H84 uCosminexus Client 08-50

適用 OS : Windows Server 2003 (x64) , Windows Server 2003 R2 (x64) , Windows Server 2008 x64 , Windows Server 2008 R2

P-2943-7D84 uCosminexus Application Server Standard 08-50

P-2943-7K84 uCosminexus Application Server Enterprise 08-50

P-2943-7S84 uCosminexus Service Platform 08-50

適用 OS : AIX 5L V5.3 , AIX V6.1

P-1M43-7D81 uCosminexus Application Server Standard 08-50

P-1M43-7K81 uCosminexus Application Server Enterprise 08-50

P-1M43-7S81 uCosminexus Service Platform 08-50

適用 OS : HP-UX 11i V2 (IPF) , HP-UX 11i V3 (IPF)

P-1J43-7D81 uCosminexus Application Server Standard 08-50

P-1J43-7K81 uCosminexus Application Server Enterprise 08-50

P-1J43-7S81 uCosminexus Service Platform 08-50

適用 OS : Red Hat Enterprise Linux AS 4 (x86) , Red Hat Enterprise Linux 5 Advanced Platform (x86) , Red Hat Enterprise Linux ES 4 (x86) , Red Hat Enterprise Linux 5 (x86) , Red Hat Enterprise Linux AS 4 (AMD64 & Intel EM64T) , Red Hat Enterprise Linux 5 Advanced Platform (AMD/Intel 64) , Red Hat Enterprise Linux ES 4 (AMD64 & Intel EM64T) , Red Hat Enterprise Linux 5 (AMD/Intel 64)

P-9S43-7D81 uCosminexus Application Server Standard 08-50

P-9S43-7K81 uCosminexus Application Server Enterprise 08-50

P-9S43-7M81 uCosminexus Web Redirector 08-50

P-9S43-7S81 uCosminexus Service Platform 08-50

適用 OS : Red Hat Enterprise Linux AS 4 (IPF) , Red Hat Enterprise Linux 5 (Intel Itanium) , Red Hat Enterprise Linux 5 Advanced Platform (Intel Itanium)

P-9V43-7D81 uCosminexus Application Server Standard 08-50

P-9V43-7K81 uCosminexus Application Server Enterprise 08-50

P-9V43-7S81 uCosminexus Service Platform 08-50

印の製品については、サポート時期をご確認ください。

これらのプログラムプロダクトのほかにもこのマニュアルをご利用になれる場合があります。詳細は「リリースノート」でご確認ください。

本製品では日立トレース共通ライブラリをインストールします。

輸出時の注意

本製品を輸出される場合には、外国為替および外国貿易法ならびに米国の輸出管理関連法規などの規制をご確認の上、必要な手続きをお取りください。

なお、ご不明な場合は、弊社担当営業にお問い合わせください。

商標類

AIX は、米国およびその他の国における International Business Machines Corporation の商標です。

AIX 5L は、米国およびその他の国における International Business Machines Corporation の商標です。

AMD は、Advanced Micro Devices, Inc. の商標です。

BIG-IP、3-DNS、iControl Services Manager、FirePass および F5 は F5 Networks Inc の商標、または登録商標です。

Borland のブランド名および製品名はすべて、米国 Borland Software Corporation の米国およびその他の国における商標または登録商標です。

CORBA は、Object Management Group が提唱する分散処理環境アーキテクチャの名称です。

HP-UX は、米国 Hewlett-Packard Company のオペレーティングシステムの名称です。

IIOP は、OMG 仕様による ORB(Object Request Broker) 間通信のネットワークプロトコルの名称です。

Internet Explorer は、米国 Microsoft Corporation の米国及びその他の国における登録商標または商標です。

Itanium は、アメリカ合衆国およびその他の国における Intel Corporation の商標です。

Java は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

JDK は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

JSP は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。

Microsoft は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Microsoft Internet Information Services は、米国 Microsoft Corp. の商品名称です。

OMG、CORBA、IIOP、UML、Unified Modeling Language、MDA、Model Driven Architecture は、Object Management Group, Inc. の米国及びその他の国における登録商標または商標です。

ORACLE は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Oracle 及び Oracle Database 11g は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Oracle 10g は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Oracle9i は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Red Hat は、米国およびその他の国で Red Hat, Inc. の登録商標若しくは商標です。

Solaris は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または

商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標がついた製品は、米国 Sun Microsystems, Inc. が開発したアーキテクチャに基づくものです。

Sun は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Sun Microsystems は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

UNIX は、The Open Group の米国ならびに他の国における登録商標です。

VMware は、VMware, Inc. の米国および各国での登録商標または商標です。

Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows Server は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows Vista は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

X/Open は、The Open Group の英国ならびに他の国における登録商標です。

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

マイクロソフト製品の表記について

このマニュアルでは、マイクロソフト製品の名称を次のように表記しています。

製品名	表記	
Microsoft(R) Internet Information Services 6.0	Microsoft IIS 6.0 または Microsoft IIS	
Microsoft(R) Windows(R) 7 Enterprise	Windows 7	Windows
Microsoft(R) Windows(R) 7 Professional		
Microsoft(R) Windows(R) 7 Ultimate		
Microsoft(R) Windows Server(R) 2003 , Enterprise Edition Operating System (x86) 日本語版	Windows Server 2003 Enterprise Edition	Windows Server 2003
Microsoft(R) Windows Server(R) 2003 , Standard Edition Operating System (x86) 日本語版	Windows Server 2003 Standard Edition	
Microsoft(R) Windows Server(R) 2003 R2 , Enterprise Edition Operating System (x86) 日本語版	Windows Server 2003 R2 Enterprise Edition	Windows Server 2003 R2
Microsoft(R) Windows Server(R) 2003 R2 , Standard Edition Operating System (x86) 日本語版	Windows Server 2003 R2 Standard Edition	
Microsoft(R) Windows Server(R) 2003 , Enterprise x64 Edition Operating System 日本語版	Windows Server 2003 Enterprise x64 Edition	Windows Server 2003 (x64)
Microsoft(R) Windows Server(R) 2003 , Standard x64 Edition Operating System 日本語版	Windows Server 2003 Standard x64 Edition	

製品名	表記	
Microsoft(R) Windows Server(R) 2003 R2 , Enterprise x64 Edition Operating System 日本語 版	Windows Server 2003 R2 Enterprise x64 Edition	Windows Server 2003 R2 (x64)
Microsoft(R) Windows Server(R) 2003 R2 , Standard x64 Edition Operating System 日本語 版	Windows Server 2003 R2 Standard x64 Edition	
Microsoft(R) Windows Server(R) 2008 Enterprise 32-bit 日本語版	Windows Server 2008 x86	
Microsoft(R) Windows Server(R) 2008 Standard 32-bit 日本語版		
Microsoft(R) Windows Server(R) 2008 Enterprise x64 Edition 日本語版	Windows Server 2008 x64	
Microsoft(R) Windows Server(R) 2008 Standard x64 Edition 日本語版		
Microsoft(R) Windows Server(R) 2008 R2 Enterprise 日本語版	Windows Server 2008 R2	
Microsoft(R) Windows Server(R) 2008 R2 Standard 日本語版		
Microsoft(R) Windows Vista(R) Business	Windows Vista Business	Windows Vista
Microsoft(R) Windows Vista(R) Enterprise	Windows Vista Enterprise	
Microsoft(R) Windows Vista(R) Ultimate	Windows Vista Ultimate	
Microsoft(R) Windows(R) XP Professional Operating System	Windows XP	
Microsoft(R) Windows(R) Internet Explorer(R)	Internet Explorer	

発行

2008年11月(第1版) 3020-3-U14

2010年9月(第2.1版) 3020-3-U14-21

著作権

All Rights Reserved. Copyright (C) 2008, 2010, Hitachi, Ltd.

変更内容

変更内容 (3020-3-U14-21)

追加・変更内容	変更箇所
<p>同時実行スレッド数の制御単位が明確になるように記述を改善した。</p>	-
<p>[訂正前] Cosminexus Web Services - Base</p> <p>[訂正後] SOAP 通信基盤</p>	<p>表 4-11 詳細レベルのトレース情報を取得する場合の指定値 (cprfstart コマンド)</p> <p>インデックス 1</p>
<p>[訂正前] Cosminexus Web Services - Base</p> <p>[訂正後] JAX-WS エンジン</p>	<p>表 4-11 詳細レベルのトレース情報を取得する場合の指定値 (cprfstart コマンド)</p> <p>インデックス 2</p>
<p>[訂正前] タイムアウト間隔 (単位 : 秒) の初期値を指定します。コマンドを実行から再度要求を出すまでに CJMSP ブローカーからの応答を待つ時間の初期値です。リトライ間隔はタイムアウト間隔の初期値の倍数が設定されます。デフォルト値は 10 秒です。また、-1 を指定した場合も、デフォルト値が適用されます。</p> <p>[訂正後] タイムアウト間隔 (単位 : 秒) の初期値を 0 ~ 2147483647 の整数で指定します。コマンドを実行から再度要求を出すまでに CJMSP ブローカーからの応答を待つ時間の初期値です。リトライ間隔はタイムアウト間隔の初期値の倍数が設定されます。デフォルト値は 10 秒です。また、0 を指定した場合は、タイムアウトは発生しません。</p>	<p>表 5-6 コマンドユーティリティの共通オプションの一覧</p> <p>-rtm timeoutInterval</p>
<p>[訂正前]</p> <ul style="list-style-type: none"> 予約語である装置名は指定できません。 (例) AUX, CON, NUL, PRN, COM1- COM9, LPT1-LPT9 指定できる文字数は、OS で規定されたディレクトリ作成時の最大文字数と同じです。 英数字と特殊文字を組み合わせて指定します。ただし、次に示す特殊文字は使用できません。 ¥ / : * ? " < > <p>[訂正後]</p> <ul style="list-style-type: none"> Windows 環境の場合は、予約デバイス名は指定できません。 (例) AUX, CON, NUL, PRN, COM1- COM9, LPT1-LPT9 指定できる文字数は、OS で規定されたディレクトリ作成時の最大文字数と同じです。 英数字と特殊文字を組み合わせて指定します。ただし、Windows 環境の場合では、次に示す特殊文字は使用できません。 ¥ / : * ? " < > <p>UNIX 環境の場合では、使用可能な文字はファイルシステムの仕様に依存します。</p>	<p>5.3 CJMSP ブローカー管理のコマンドの詳細</p> <p>cjmsbroker (CJMSP ブローカーの起動) -name < インスタンス名称 ></p>

追加・変更内容	変更箇所
<p>[訂正前]</p> <ul style="list-style-type: none"> 予約語である装置名は指定できません。指定した場合、エラーメッセージが表示されます。 <p>(例)</p> <p>AUX, CON, NUL, PRN, COM1- COM9, LPT1-LPT9</p> <ul style="list-style-type: none"> 英数字と特殊文字を組み合わせて指定します。ただし、次に示す特殊文字は使用できません。使用できない特殊文字を使用した場合、エラーメッセージが表示されます。 <p>¥ / : * ? " < > </p> <p>[訂正後]</p> <ul style="list-style-type: none"> Windows 環境の場合は、予約デバイス名は指定できません。指定した場合、エラーメッセージが表示されます。 <p>(例)</p> <p>AUX, CON, NUL, PRN, COM1- COM9, LPT1-LPT9</p> <ul style="list-style-type: none"> 英数字と特殊文字を組み合わせて指定します。ただし、Windows 環境の場合では、次に示す特殊文字は使用できません。 <p>¥ / : * ? " < > </p> <p>UNIX 環境の場合では、使用可能な文字はファイルシステムの仕様に依存します。使用できない特殊文字を使用した場合、エラーメッセージが表示されず。</p>	<p>5.3 CJMSP ブローカー管理のコマンドの詳細</p> <p>cjmsbroker (CJMSP ブローカーの起動) -varhome < ロケーション ></p>
<p>[訂正前]</p> <p>コマンドを実行してから停止されるまでの待機時間を秒単位で指定します。</p> <p>[訂正後]</p> <p>コマンドを実行してから停止されるまでの待機時間を秒単位で指定します。範囲は 0 ~ 2147483647 の整数で指定します。</p>	<p>5.3 CJMSP ブローカー管理のコマンドの詳細</p> <p>cjmsicmd shutdown bkr (CJMSP ブローカーの停止) 引数 -time < 待機時間 ></p>
<p>[追加]</p> <p>注意事項</p> <p>仮想化システムで使用するコマンドを実行している間は、[Ctrl] + [C] キーなどで外部から停止しないでください。</p>	<p>9.3 仮想化システムで使用するコマンドの詳細</p> <p>表 9-3 の直後</p>

変更内容 (3020-3-U14-20) uCosminexus Application Server Enterprise 08-50 , uCosminexus Application Server Standard 08-50 , uCosminexus Client 08-50 , uCosminexus Developer Professional 08-50 , uCosminexus Developer Standard 08-50 , uCosminexus Service Architect 08-50 , uCosminexus Service Platform 08-50 , uCosminexus Web Redirector 08-50

追加・変更内容	変更箇所
Cosminexus JMS プロバイダで使用するコマンドを追加した。	1.1, 5 章
仮想化システムで使用するコマンドを追加した。	1.1, 1.4, 9 章, 付録 J.1

追加・変更内容	変更箇所
次の製品の適用 OS に Windows Server 2008 R2 を追加した。 <ul style="list-style-type: none"> • uCosminexus Application Server Standard • uCosminexus Application Server Enterprise • uCosminexus Web Redirector • uCosminexus Service Platform • uCosminexus Client 	1.4 , 2.2 , 2.3 , 2.4 , 2.5 , 3.3 , 14.2 , 17.2
次の製品の適用 OS に Windows 7 を追加した。 <ul style="list-style-type: none"> • uCosminexus Developer Standard • uCosminexus Developer Professional • uCosminexus Service Architect • uCosminexus Client 	1.4 , 2.2 , 2.3 , 2.4 , 2.5 , 3.3 , 14.2 , 17.2
JDK 6 に対応した。	10 章 , 15.2
Jheapprof コマンド (日立クラス別統計情報付き拡張スレッドダンプの出力) に , 次の引数を追加した。 <ul style="list-style-type: none"> • -garbage • -nogarbage • -rootobjectinfo • -norootobjectinfo • -rootobjectinfost 	10.2
jheapprofanalyzer コマンド (日立クラス別統計情報解析ファイルの出力) を追加した。	10.2
次の製品の適用 OS から Solaris を削除した。 <ul style="list-style-type: none"> • uCosminexus Application Server Standard • uCosminexus Application Server Enterprise 	-

単なる誤字・脱字などはお断りなく訂正しました。

なお、「はじめに」の記載の一部を「このマニュアルの参考情報」に移動しました。

はじめに

このマニュアルは、Cosminexus（コズミネクサス）のアプリケーションサーバの機能について説明したものです。このマニュアルでは、システムの構築、運用およびアプリケーションの開発で使用するコマンドについて説明します。

アプリケーションサーバでは、次に示すプログラムプロダクトを使用してシステムを構築、運用します。

- uCosminexus Application Server Enterprise
- uCosminexus Application Server Standard
- uCosminexus Client
- uCosminexus Developer Professional
- uCosminexus Developer Standard
- uCosminexus Service Architect
- uCosminexus Service Platform
- uCosminexus Web Redirector

このマニュアルでは、これらのプログラムプロダクトの構成ソフトウェアのうち、次に示す構成ソフトウェアについて説明しています。

- Cosminexus Component Container
- Cosminexus Component Container - Client
- Cosminexus Component Container - Redirector
- Cosminexus Component Transaction Monitor
- Cosminexus DABroker Library
- Cosminexus Developer's Kit for Java
- Cosminexus Performance Tracer
- Cosminexus TPBroker

なお、オペレーティングシステム（OS）の種類によって、機能が異なる場合があります。

対象読者

このマニュアルは、アプリケーションサーバを使用したシステムを設計、構築または運用する方、およびアプリケーションサーバで動作するアプリケーションを開発する方を対象としています。

なお、次の内容を理解されていることを前提としています。

システムを設計、構築または運用する方

- Windows またはご使用の UNIX のシステム構築および運用に関する知識
- Java EE に関する知識
- SQL およびリレーショナルデータベースに関する基本的な知識
- CORBA に関する基本的な知識

アプリケーションを開発する方

はじめに

- Windows の基本操作に関する知識
- Java によるプログラム開発に関する基本的な知識
- 使用する IDE に関する基本的な知識

ご利用の製品ごとの用語の読み替えについて

ご利用の製品によっては、マニュアルで使用している用語を、ご利用の製品名に読み替える必要があります。

次の表に従って、マニュアルで使用している用語をご利用の製品名に読み替えてください。

ご利用の製品名	マニュアルで使用している用語
uCosminexus Developer Professional ¹	Application Server および Application Server Enterprise
uCosminexus Developer Standard ^{1, 2}	Application Server
uCosminexus Service Architect ¹	Application Server および Application Server Enterprise
uCosminexus Service Platform	

注 1 テスト環境で使用している場合にだけ読み替えが必要です。

注 2 uCosminexus Developer Standard と Application Server には一部機能差があります。機能差については、マニュアル「Cosminexus アプリケーションサーバ V8 アプリケーション開発ガイド」の Developer Standard 使用時の注意事項に関する説明を参照してください。

文法で使用している記号

このマニュアルの文法で使用している記号について次に示します。

記述記号

記述記号	意味
	横に並べられた複数の項目に対する項目間の区切りを示し、「または」を意味します。 (例) A B A または B を指定することを示します。
{ }	この記号で囲まれている複数の項目のうちから一つを選択することを示します。項目が横に並べられ、記号 で区切られている場合は、そのうちの一つを選択します。 (例) {A B C} A, B または C のどれかを指定することを示します。
[]	この記号で囲まれている項目は省略してもよいことを示します。複数の項目が横に並べて記述されている場合には、すべてを省略するか、記号 { } と同じくどれか一つを選択します。 (例 1) [A] 「何も指定しない」か「A を指定する」ことを示します。 (例 2) [B C] 「何も指定しない」か「B または C を指定する」ことを示します。

記述記号	意味
...	記述が省略されていることを示します。 (例) ABC... ABC の後ろに記述があり、その記述が省略されていることを示します。
< >	この記号で囲まれている項目は、該当する要素やファイルなどを指定したり、該当する要素が表示されたりすることを示します。 (例 1) <プロパティ> プロパティを記述します。またはプロパティが表示されます。 (例 2) <ファイル名> ファイル名を指定します。
...	この記号の直前に示す記号を繰り返し、複数個指定できることを示します。 (例) <プロパティ>... プロパティは複数個、繰り返して指定できます。

構文要素

構文要素	定義
英字	A ~ Z a ~ z
英小文字	a ~ z
英大文字	A ~ Z
数字	0 ~ 9
英数字	A ~ Z a ~ z 0 ~ 9
記号	! " # \$ % & ' () + , _ . / : ; < = > @ [] ^ - { } タブ 空白

注 すべての半角文字を使用してください。

目次

1	コマンドの概要	1
1.1	コマンドの種類	2
1.2	コマンドの入力形式	6
1.3	コマンドの文法の記述形式	8
1.4	コマンド使用時の注意事項	9
2	J2EE サーバで使用するコマンド	15
2.1	J2EE サーバで使用するコマンドの概要	16
2.2	J2EE サーバを操作するコマンド	18
2.3	J2EE アプリケーションで使用するコマンド	42
2.4	J2EE サーバで使用するリソース操作コマンド	90
2.5	J2EE サーバで使用するそのほかのコマンド	153
3	バッチサーバで使用するコマンド	175
3.1	バッチサーバで使用するコマンドの概要	176
3.2	バッチサーバを操作するコマンド	178
3.3	バッチアプリケーションで使用するコマンド	183
3.4	バッチサーバで使用するリソース操作コマンド	194
3.5	バッチサーバで使用するそのほかのコマンド	196
4	性能解析トレースで使用するコマンド	197
4.1	性能解析トレースで使用するコマンドの一覧	198
4.2	性能解析トレースで使用するコマンドの詳細	199
5	Cosminexus JMS プロバイダで使用するコマンド	221
5.1	Cosminexus JMS プロバイダで使用するコマンドの概要	222
5.2	Cosminexus JMS プロバイダで使用するコマンドの一覧	223
5.2.1	CJMSP ブローカーユーティリティのコマンドの一覧	223
5.2.2	コマンドユーティリティのコマンドの一覧	223
5.2.3	Cosminexus JMS プロバイダで使用するコマンドの格納先	225

5.3	CJMSP ブローカー管理のコマンドの詳細	226
5.4	コネクションサービス管理のコマンドの詳細	234
5.5	物理的送信先管理のコマンドの詳細	236
5.6	永続化サブスクリバ管理のコマンドの詳細	245
5.7	物理的送信先の設定プロパティ	248
5.8	Cosminexus JMS プロバイダで使用するコマンドの注意事項	249

6

CTM で使用するコマンド	251
6.1 CTM で使用するコマンドの一覧	252
6.2 CTM で使用するコマンドの詳細	254

7

Management Server で使用するコマンド	305
7.1 Management Server で使用するコマンドの一覧	306
7.2 Management Server で使用するコマンドの詳細	307
7.3 mngsvrutil コマンドのサブコマンドの詳細	332

8

Smart Composer 機能で使用するコマンド	353
8.1 Smart Composer 機能で使用するコマンドの一覧	354
8.1.1 J2EE アプリケーションを実行するシステムで使用できるコマンドの一覧	354
8.1.2 パッチアプリケーションを実行するシステムで使用できるコマンドの一覧	355
8.1.3 旧バージョンとの互換用のコマンドの一覧	356
8.2 コマンドを同時に複数実行した場合の排他制御	359
8.3 Smart Composer 機能で使用するコマンドの詳細	362

9

仮想化システムで使用するコマンド	419
9.1 仮想化システムで使用するコマンドの一覧	420
9.2 コマンド実行時の排他制御	422
9.3 仮想化システムで使用するコマンドの詳細	423

10

JavaVM で使用するコマンド	443
10.1 JavaVM で使用するコマンドの一覧	444
10.2 JavaVM で使用するコマンドの詳細	445

11	ログの運用で使用するコマンド	471
11.1	ログの運用で使用するコマンドの一覧	472
11.2	ログの運用で使用するコマンドの詳細	473
12	監査ログの設定で使用するコマンド	475
12.1	監査ログの設定で使用するコマンドの一覧	476
12.2	監査ログの設定で使用するコマンドの詳細	477
13	統合ユーザ管理で使用するコマンド	479
13.1	統合ユーザ管理で使用するコマンドの一覧	480
13.2	統合ユーザ管理で使用するコマンドの詳細	481
14	Java アプリケーションで使用するコマンド	491
14.1	Java アプリケーションで使用するコマンドの一覧	492
14.2	Java アプリケーションで使用するコマンドの詳細	493
15	Web アプリケーションで使用するコマンド	501
15.1	Web アプリケーションで使用するコマンドの一覧	502
15.2	Web アプリケーションで使用するコマンドの詳細	503
16	構築済み実行環境の切り替えで使用するコマンド	513
16.1	構築済み実行環境の切り替えで使用するコマンドの一覧	514
16.2	構築済み実行環境の切り替えで使用するコマンドの詳細	515
17	Web コンテナサーバで使用するコマンド	521
17.1	Web コンテナサーバで使用するコマンドの一覧	522
17.2	Web コンテナサーバで使用するコマンドの詳細	523
付録		543
付録 A	JavaVM で使用するポート番号と設定変更方法	544

付録 B dabsetup (Cosminexus DABroker Library のセットアップ)	545
付録 C J2EE アプリケーションを実行するシステムの構築と削除	547
付録 C.1 システム構築の流れ	547
付録 C.2 J2EE サーバのセットアップ	551
付録 C.3 システムの削除の流れ	554
付録 C.4 J2EE サーバのアンセットアップ	554
付録 D J2EE アプリケーションを実行するシステムの起動と停止 (Windows の場合)	556
付録 D.1 システムの起動手順	556
付録 D.2 システムの起動方法	561
付録 D.3 システムの停止手順	569
付録 D.4 システムの停止方法	571
付録 E J2EE アプリケーションを実行するシステムの起動と停止 (UNIX の場合)	576
付録 E.1 システムの起動手順	576
付録 E.2 システムの起動方法	581
付録 E.3 システムの停止手順	590
付録 E.4 システムの停止方法	592
付録 F バッチアプリケーションを実行するシステムの構築と削除	598
付録 F.1 システム構築の流れ	598
付録 F.2 バッチサーバのセットアップ	602
付録 F.3 システムの削除の流れ	604
付録 F.4 バッチサーバのアンセットアップ	604
付録 G バッチアプリケーションを実行するシステムの起動と停止 (Windows の場合)	606
付録 G.1 システムの起動手順	606
付録 G.2 システムの起動方法	607
付録 G.3 システムの停止手順	609
付録 G.4 システムの停止方法	610
付録 H バッチアプリケーションを実行するシステムの起動と停止 (UNIX の場合)	612
付録 H.1 システムの起動手順	612
付録 H.2 システムの起動方法	613
付録 H.3 システムの停止手順	615
付録 H.4 システムの停止方法	616
付録 I Smart Composer 機能で使用できる抽象パラメタ	618
付録 I.1 抽象パラメタの展開	618
付録 I.2 抽象パラメタによって展開される実パラメタ	619
付録 J このマニュアルの参考情報	625

付録 J.1 関連マニュアル	625
付録 J.2 このマニュアルでの表記	628
付録 J.3 英略語	631
付録 J.4 常用漢字以外の漢字の使用について	633
付録 J.5 KB (キロバイト) などの単位表記について	633

索引

635

1

コマンドの概要

この章では、アプリケーションサーバで使用するコマンドの種類、入力形式、文法などの記述形式について説明します。

-
- 1.1 コマンドの種類
 - 1.2 コマンドの入力形式
 - 1.3 コマンドの文法の記述形式
 - 1.4 コマンド使用時の注意事項
-

1.1 コマンドの種類

ここでは、アプリケーションサーバで使用するコマンドの種類について説明します。

アプリケーションサーバでは、実行するアプリケーションの種類によって使用できるコマンドが異なります。この節では、コマンドを次の四つに分類して説明します。

J2EE アプリケーションを実行するシステムで使用するコマンド

バッチアプリケーションを実行するシステムで使用するコマンド

Java アプリケーションを実行するシステムで使用するコマンド

Web アプリケーションを実行するシステムで使用するコマンド

それぞれのコマンドについて次の表に示します。

表 1-1 J2EE アプリケーションを実行するシステムで使用するコマンド

コマンドの種類	コマンドの説明	参照先
J2EE サーバで使用するコマンド	J2EE サーバの起動や停止、J2EE アプリケーションの設定などで使用する、cj で始まるコマンドです。 例：cjstartsv, cjenvupdate	2 章
性能解析トレースで使用するコマンド	性能解析トレースでアプリケーションサーバのシステムの処理性能を解析する場合に、PRF デーモンの起動や停止、トレース情報の出力などで使用する、cprf で始まるコマンドです。 例：cprfstart, cprfed	4 章
Cosminexus JMS プロバイダで使用するコマンド	Cosminexus JMS プロバイダを使用して、メッセージの送受信などをするコマンドです。	5 章
CTM で使用するコマンド	CTM を利用してリクエストのスケジューリングや負荷分散をする場合に、CTM ドメインの起動や停止、CTM の情報の出力、キューの閉塞などで使用する、ctm で始まるコマンドです。 例：ctmstart, ctmholdque	6 章
Management Server で使用するコマンド	Management Server を使用する場合に、運用管理エージェントの稼働確認、起動や停止、Management Server の起動や停止などで使用するコマンドです。 例：adminagentcheck, mngsvrctl	7 章

コマンドの種類	コマンドの説明	参照先
Smart Composer 機能で使用するコマンド	Smart Composer 機能を使用する場合に、Management Server の管理ユーザアカウントの設定、Management Server 上の Web システムの構築などで使用するコマンドです。 例：cmx_admin_passwd, cmx_build_system	8 章
仮想化システムで使用するコマンド	仮想サーバマネージャから情報を取得したり、更新したりするコマンドです。	9 章
JavaVM で使用するコマンド	JavaVM の core ファイルの取得、ガーベージコレクションの発生などで使用するコマンドです。 例：javacore, javagc	10 章
ログの運用で使用するコマンド	アプリケーションサーバのログを収集する場合に、snapshot ログの収集などで使用するコマンドです。 例：snapshotlog	11 章
監査ログの設定で使用するコマンド	監査ログのセットアップ時に使用するコマンドです。 例：auditsetup	12 章
統合ユーザ管理で使用するコマンド	統合ユーザ管理機能を使用する場合に、ユーザ情報の暗号化、暗号鍵ファイルの作成などで使用するコマンドです。 例：convpw, ssogenkey	13 章
構築済み実行環境の切り替えで使用するコマンド	アプリケーションサーバのシステム環境の退避・回復などで使用するコマンドです。 例：cosmienv	16 章

注 CTM は、構成ソフトウェアに Cosminexus Component Transaction Monitor を含む製品だけで利用できます。利用できる製品については、マニュアル「Cosminexus アプリケーションサーバ概説」を参照してください。

表 1-2 バッチアプリケーションを実行するシステムで使用するコマンド

コマンドの種類	コマンドの説明	参照先
バッチサーバで使用するコマンド	バッチサーバの起動や停止、バッチアプリケーションの実行や強制停止などで使用する、cj で始まるコマンドです。 例：cjstartsv, cjexecjob	3 章
性能解析トレースで使用するコマンド	性能解析トレースでアプリケーションサーバのシステムの処理性能を解析する場合に、PRF デーモンの起動や停止、トレース情報の出力などで使用する、cprf で始まるコマンドです。 例：cprfstart, cprfed	4 章

1. コマンドの概要

コマンドの種類	コマンドの説明	参照先
CTM で使用するコマンド	CTM を利用してリクエストのスケジューリングや負荷分散をする場合に、CTM ドメインの起動や停止、CTM の情報の出力、キューの閉塞などで使用する、ctm で始まるコマンドです。 例：ctmstart, ctmholdque	6 章
Management Server で使用するコマンド	Management Server を使用する場合に、運用管理エージェントの稼働確認、起動や停止、Management Server の起動や停止などで使用するコマンドです。 例：adminagentcheck, mngsvrctl	7 章
Smart Composer 機能で使用するコマンド	Smart Composer 機能を使用する場合に、Management Server の管理ユーザアカウントの設定、Management Server 上の Web システムの構築などで使用するコマンドです。 例：cmx_admin_passwd, cmx_build_system	8 章
JavaVM で使用するコマンド	JavaVM の core ファイルの取得、ガーベージコレクションの発生などで使用するコマンドです。 例：javacore, javagc	10 章
ログの運用で使用するコマンド	アプリケーションサーバのログを収集する場合に、snapshot ログの収集などで使用するコマンドです。 例：snapshotlog	11 章
監査ログの設定で使用するコマンド	監査ログのセットアップ時に使用するコマンドです。 例：auditsetup	12 章
構築済み実行環境の切り替えで使用するコマンド	アプリケーションサーバのシステム環境の退避・回復などで使用するコマンドです。 例：cosmienv	16 章

注 CTM は、構成ソフトウェアに Cosminexus Component Transaction Monitor を含む製品だけで利用できます。利用できる製品については、マニュアル「Cosminexus アプリケーションサーバ概説」を参照してください。

表 1-3 Java アプリケーションを実行するシステムで使用するコマンド

コマンドの種類	コマンドの説明	参照先
Java アプリケーションで使用するコマンド	Java アプリケーションの起動、ログの出力先サブディレクトリの削除などで使用するコマンドです。 例：ejclstartap, ejcldellog	14 章

表 1-4 Web アプリケーションを実行するシステムで使用するコマンド

コマンドの種類	コマンドの説明	参照先
Web アプリケーションで使用するコマンド	JSP ソースのチェックなどで使用するコマンドです。 例：cjsp2java	15 章

参考

サーブレットエンジンモードで使用するコマンドの種類を次に示します。

- 性能解析トレースで使用するコマンド
- JavaVM で使用するコマンド
- ログの運用で使用するコマンド
- 統合ユーザ管理で使用するコマンド
- Web アプリケーションで使用するコマンド
- Web コンテナサーバで使用するコマンド

注

サーブレットエンジンモードだけで使用できるコマンドです。

サーブレットエンジンモードで Web コンテナサーバを使用する場合に、Web コンテナサーバの起動や停止、プール管理情報の設定などで使用します。

例：cjstartweb, cjwebeditpool

コマンドについては、「17. Web コンテナサーバで使用するコマンド」を参照してください。

1.2 コマンドの入力形式

コマンドの入力形式を次に示します。

コマンド名称 [オプション...]

各項目について説明します。なお、コマンドプロンプトを「\$」、コマンド名称を「cmd」と表記します。

(1) コマンド名称

実行するコマンドのファイル名を指定します。

空白を含むパスを指定してコマンドを実行する場合、パス全体を "" で囲む必要があります。

Windows の場合

- 誤った指定例：\$ C:¥Program Files¥Hitachi¥Cosminexus¥CC¥server¥bin¥cmd
- 正しい指定例：\$ "C:¥Program Files¥Hitachi¥Cosminexus¥CC¥server¥bin¥cmd"

UNIX の場合

- 誤った指定例：\$ /opt/program path/bin/command
- 正しい指定例：\$ "/opt/program path/bin/command"

(2) 引数

引数には、オプションも含まれます。オプションの入力形式および指定規則を次に示します。

(a) オプションの入力形式

オプションは、「-」(ハイフン)で始まる文字列です。オプションの入力形式には、オプション引数を指定しない形式と、1個のオプション引数を指定する形式があります。

オプション引数を指定しない形式

\$ cmd - オプションフラグ

1個のオプション引数を指定する形式

\$ cmd - オプションフラグ <スペースまたはタブ> オプション引数

(凡例)

- オプションフラグ
1文字の半角英数字です。大文字と小文字が区別されます。
- オプション引数
オプションフラグに対する引数です。

(b) オプションの指定規則

- オプションフラグは、一つの「-」(ハイフン)にまとめて指定できません。
誤った指定例: `$ cmd -abc`
正しい指定例: `$ cmd -a -b -c`
- オプション引数を必要とするオプションフラグのオプション引数は、省略できません。
- オプションフラグとオプション引数の間には、スペースまたはタブが必要です。
誤った指定例: `$ cmd -afile`
正しい指定例: `$ cmd -a file`
- 同じオプションフラグは、複数指定できません。
誤った指定例: `$ cmd -a 1 -a 2`
- オプション引数に空白を含む場合、オプション引数全体を "" で囲む必要があります。
誤った指定例: `$ cmd -a file 1`
正しい指定例: `$ cmd -a "file 1"`

1.3 コマンドの文法の記述形式

2章～15章では、コマンドの文法について次の形式で説明します。なお、各コマンドは、アルファベットの順に説明します。

形式

コマンドの入力形式を示します。

機能

コマンドの機能について説明します。

引数

コマンドの引数およびオプションについて説明します。

出力形式

コマンドの出力形式を示します。

入力例・出力例

コマンドの入力例および出力例を示します。

戻り値

コマンドの戻り値について説明します。

注意事項

コマンドを実行する上での注意事項について説明します。

参考

各コマンドの説明では、上記の項目のうち必要な項目についてだけ説明しています。また、上記の項目以外に、各コマンドの固有情報を記載している場合があります。

1.4 コマンド使用時の注意事項

コマンド使用時の注意事項を説明します。

(1) Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista でアプリケーションサーバが提供するコマンドを使用する場合の注意事項

アプリケーションサーバが提供するコマンドは、管理者特権で実行する必要があります。「管理者：コマンドプロンプト」で実行してください。

「管理者：コマンドプロンプト」は、Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista で提供されている機能を使用して起動してください。起動方法の例を次に示します。

1. [スタート] ボタンをクリックします。
2. [すべてのプログラム] - [アクセサリ] を選択します。
3. [コマンド プロンプト] を右クリックして、[管理者として実行] をクリックします。

管理者のパスワードまたは確認を求められた場合は、画面の指示に従って、パスワードを入力するか、または確認情報を設定してください。

(2) サーバ管理コマンドを使用する場合の注意事項

コマンドの実行でメモリ不足が発生しコマンドが正常に実行できない場合は、サーバ管理コマンド用のオプション定義ファイル (usrconf.bat) 中の USRCONF_JVM_ARGS キーに JavaVM のオプション `-Xmx<size>` を指定して、メモリ割り当てプールの最大サイズを拡張することで対応します。`-Xmx<size>` の指定がない場合のデフォルト値は 64MB です。

[例]

Windows の場合

```
set USRCONF_JVM_ARGS=-Xmx512m
```

UNIX の場合

```
set USRCONF_JVM_ARGS="-Xmx512m"
```

属性ファイルの XML 宣言部分で指定されているエンコーディングは、該当する属性ファイルのエンコーディングを決定するものであり、サーバ側の Deployment Descriptor ファイルのエンコーディングには影響しません (サーバ側の Deployment Descriptor ファイルのエンコーディングは必ず「UTF-8」となります)。

Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista を使用している場合に、管理者特権でコマンドプロ

1. コマンドの概要

ンプトを開いていないときには、管理者特権エラーとなり、終了コード「9」が返されます。

UNIX の場合、Component Container 管理者を設定すれば、スーパーユーザ以外の一般ユーザでもコマンドを実行できます。

Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista では Administrator 権限を持つユーザであっても、コマンドプロンプトを管理者特権で開いていない場合、コマンドを実行できません。

ディレクトリ名、ファイル名に Unicode の補助文字は使用できません。

ログファイルに、実行時の文字エンコーディングに変換できない文字を出力した場合、「?」に変換されます。

Bean のメソッドに設定した実行権限が有効となるのは、create メソッド、ビジネスメソッド、および remove メソッドだけです。これら以外のメソッドに対してサーバ管理コマンドで設定された実行権限は無視されます。

コマンドで J2EE サーバ、またはバッチサーバリモート接続を実行した場合、コンソールおよびダイアログに KDJE30045-E のメッセージが表示されることがあります。その場合には、サーバ管理コマンドと J2EE サーバ、またはバッチサーバの Cosminexus Component Container のバージョンを合わせてください。

Cosminexus Component Container 06-00 以前のバージョンからリモートで cjresetsv コマンドを実行すると KDJE37516-I のメッセージが表示されますが、サーバ管理コマンドの排他情報はリセットされません。

削除系コマンドと一覧表示系コマンドを同一リソースに対し、同時に実行すると、J2EE サーバ、またはバッチサーバを起動したコンソールに KDJE39057-E のメッセージが表示されることがありますが、J2EE サーバ、またはバッチサーバの動作への影響はありません。

削除系コマンドと一覧表示系コマンドを次に示します。

削除系サーバ管理コマンド

cjdeleteapp

cjdeleteres

cjundeployrar

cjdeletelibjar

一覧表示系サーバ管理コマンド

cjlistapp

cjlistres

cjlistrar

cjlistlibjar

サーバ管理コマンドのコマンドオプションに空白を含む場合は、次のように指定してください。

Windows の場合

空白を含むコマンドオプションをダブルクォーテーション (") で囲んでください。

(例) Display name が TP1/Message Queue - Access のリソースアダプタの情報を取得する場合

```
cjgetresprop -type rar -resname "TP1/Message Queue - Access" -c C:¥temp¥rar.xml
```

AIX の場合

空白を含むコマンドオプションをダブルクォーテーション ("") で囲んでください。

(例) Display name が TP1/Message Queue - Access のリソースアダプタの情報を取得する場合

```
cjgetresprop -type rar -resname "TP1/Message Queue - Access" -c /home/rar.xml
```

そのほかの UNIX の場合

コマンドオプション全体をダブルクォーテーション ("") で囲み、さらに空白を含むコマンドオプションをシングルクォーテーション ("") で囲んでください。

(例) Display name が TP1/Message Queue - Access のリソースアダプタの情報を取得する場合

```
cjgetresprop "-type rar -resname 'TP1/Message Queue - Access' -c /home/rar.xml"
```

ただし、次の場合には正しく動作しないことがあります。

- シングルクォーテーションで囲んだオプション中に空白が 2 個以上連続している。
- 先頭のシングルクォーテーションの直後に空白がある、または末尾のシングルクォーテーションの直前に空白がある。

サーバ管理コマンドで使用する属性ファイルでリソース環境変数用のタグ

(<resource-env-ref> タグ内の <linked-to> タグ) に指定する「キューの表示名」には、先頭がアンダースコア "_" で始まる文字列、または連続したアンダースコア "_" を含む文字列は指定しないでください。

CLASSPATH 環境変数、またはサーバ管理コマンド用オプション定義ファイルで指定したクラスパスに含まれるクラスから参照されるクラスは、アプリケーションに含めないでください。

Windows でサーバ管理コマンドを実行する場合、PATH 環境変数や CLASSPATH 環境変数に設定するパスに「^、&」を含む文字列を使用しないでください。

(3) Smart Composer 機能でコマンドを使用する場合の注意事項

次のコマンドを使用する場合は、物理ティアの J2EE サーバのコンフィグレーション設定で、J2EE サーバの運用監視を有効 (ejbserver.instrumentation.enabled パラメータに true を設定) にしてください。

- cmx_start_application
- cmx_stop_application

1. コマンドの概要

- cmx_start_resource
- cmx_stop_resource

なお、物理ティアの設定は、簡易構築定義ファイルで実施します。簡易構築定義ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」を参照してください。

（４）管理ユーザアカウントの省略機能を有効にする場合の注意事項

Manager Server および Smart Composer で使用するコマンドの中に、管理ユーザアカウントの省略機能を有効にした場合に、通常と動作が変わるコマンドがあります。一覧を次の表に示します。

表 1-5 管理ユーザアカウント省略機能有効時に動作が変わるコマンド

分類	コマンド名	管理ユーザアカウント省略機能有効時の動作
Manager Server で使用するコマンド	mngsvrctl コマンド	次に示す引数 setup の設定が無効となります。 <ul style="list-style-type: none">• -u <管理ユーザ ID>• -p <管理パスワード>• -nu <管理ユーザ ID>• -np <管理パスワード> メッセージが出力されますが、セットアップ処理は通常どおり実行します。引数 start および stop の設定は有効となります。管理ユーザアカウントおよびパスワードは、設定および変更できません。
	mngsvrutil コマンド	次に示す引数の設定が無効となります。 <ul style="list-style-type: none">• -u <管理ユーザ ID>• -p <管理パスワード>
Smart Composer で使用するコマンド	cmx_admin_passwd コマンド以外	
	cmx_admin_passwd コマンド	設定値が無効となり、メッセージが出力されます。管理ユーザアカウントおよびパスワードは、設定および変更できません。

コマンドのオプションのデフォルト値を共通定義ファイルとして設定できます。これらのファイルも管理ユーザアカウントの省略機能を有効にしたとき、動作が変わります。動作が変わるファイルの一覧を次の表に示します。

表 1-6 管理ユーザアカウント省略機能有効時に動作が変わるファイル

ファイル名	コマンド定義ファイルの種類	管理ユーザアカウント省略機能有効時の動作
.mngsvrutilrc	mngsvrutil コマンド定義ファイル	次のプロパティの設定が無効となります。 <ul style="list-style-type: none"> • mngsvrutil.connect.userid • mngsvrutil.connect.password
mngsvrutilcl.properties		
.cmxrc	Smart Composer で使用するコマンドの定義ファイル	次のプロパティの設定が無効となります。 <ul style="list-style-type: none"> • cmx.connect.userid • cmx.connect.passwd
cmxclient.properties		
.mngsvrmonitorrc	mngsvrmonitor コマンド定義ファイル	次のプロパティの設定が無効となります。 <ul style="list-style-type: none"> • mngsvrmonitor.connect.userid • mngsvrmonitor.connect.password

(5) 仮想サーバマネージャのコマンドを使用する場合の注意事項

仮想サーバマネージャのコマンドは、仮想サーバマネージャが起動している状態で実行できます。仮想サーバマネージャが起動していない状態でコマンドを実行した場合は、メッセージを出力してエラー終了します。

仮想サーバマネージャのコマンドは、Windows の場合は Administrator 権限を持つユーザが、Linux の場合は root 権限を持つユーザが実行できます。

仮想サーバマネージャのメッセージは、標準出力で出力されます。

2

J2EE サーバで使用するコマンド

この章では、J2EE サーバで使用するコマンドの入力形式、機能などについて説明します。

-
- 2.1 J2EE サーバで使用するコマンドの概要
 - 2.2 J2EE サーバを操作するコマンド
 - 2.3 J2EE アプリケーションで使用するコマンド
 - 2.4 J2EE サーバで使用するリソース操作コマンド
 - 2.5 J2EE サーバで使用するそのほかのコマンド
-

2.1 J2EE サーバで使用するコマンドの概要

ここでは、J2EE サーバで使用するコマンドの種類、プロバイダ URL について説明します。

(1) J2EE サーバで使用するコマンドの種類

J2EE サーバで使用するコマンドの種類について説明します。

J2EE サーバで使用するコマンドの種類と格納先を、次の表に示します。

表 2-1 J2EE サーバで使用するコマンドの種類と格納先

J2EE サーバで使用するコマンドの種類	コマンドの格納先
J2EE サーバを操作するコマンド	<ul style="list-style-type: none"> Windows の場合 <Cosminexus のインストールディレクトリ>\¥CC¥server¥bin¥ UNIX の場合 /opt/Cosminexus/CC/server/bin/
J2EE アプリケーションで使用するコマンド	<ul style="list-style-type: none"> Windows の場合 <Cosminexus のインストールディレクトリ>\¥CC¥admin¥bin¥
J2EE サーバで使用するリソース操作コマンド	<ul style="list-style-type: none"> UNIX の場合 /opt/Cosminexus/CC/admin/bin/

(2) プロバイダ URL について

CORBA ネーミングサービスへのアクセスプロトコル、および CORBA ネーミングサービスが稼働しているホスト名称、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

なお、該当する CORBA ネーミングサービスは、接続先 J2EE サーバがネーミングサービスとして利用しているものです。サポートしているアクセスプロトコルは corbaname だけです。

以前のバージョンで、iioploc または iiopname を指定していた場合、corbaname に読み替えられます。

iioploc または iiopname を指定する場合は、次の形式で指定します。

<プロトコル名称>://<ホスト名称>:<ポート番号>

<ホスト名称> には CORBA ネーミングサービスが稼働しているホスト名称を指定しま

す。ただし、次の場合は `vbroker.se.iiop_tp.host` キーに指定した値を <ホスト名称> に指定します。

接続先 J2EE サーバの `usrconf.properties` (J2EE サーバ用ユーザプロパティファイル) で `ejbserver.naming.startupMode=inprocess` を指定し、かつ `vbroker.se.iiop_tp.host` キーに値を指定している。

接続先 J2EE サーバの `usrconf.properties` (J2EE サーバ用ユーザプロパティファイル) で `ejbserver.naming.startupMode>manual` または `automatic` を指定し、CORBA ネーミングサービスの実行時オプションで `vbroker.se.iiop_tp.host` キーに値を指定している。

<ホスト名称> を省略した場合は次のようになります。

`usrconf.properties` (サーバ管理コマンド用システムプロパティファイル) の `"ejbserver.naming.protocol"` キーの値をアクセスプロトコルとして使用します。

`"ejbserver.naming.host"` キーの値をホスト名称として使用します。

`"ejbserver.naming.port"` キーの値をポート番号として使用します。

また、<ホスト名称> のデフォルト値は次のとおりです。

`"ejbserver.naming.protocol"` キー : `corbaname`

`"ejbserver.naming.host"` キー : `localhost`

`"ejbserver.naming.port"` キー : `900`

2.2 J2EE サーバを操作するコマンド

ここでは、J2EE サーバを操作するコマンドについて説明します。

J2EE サーバを操作するコマンドの一覧を、次の表に示します。

表 2-2 J2EE サーバを操作するコマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjdumpsv	J2EE サーバのスレッドダンプの取得	J2EE サーバで障害が発生した場合などに、JavaVM のスレッドダンプを取得します。	-
cjenvsetup	Component Container 管理者のセットアップ	インストールファイルの属性を変更して、root 権限を持つユーザ以外で運用できるようにします。このコマンドは、UNIX 用です。	-
cjenvupdate	作業ディレクトリおよびユーザ定義ファイルの移行	Cosminexus Component Container の旧環境で利用されていた作業ディレクトリおよびユーザ定義ファイルを新環境で動作する形式に移行します。	-
cjgetsysinfo	OS 状態情報の取得	OS の状態情報を取得します。	-
cjrarupdate	リソースアダプタのバージョンアップ	Cosminexus Component Container にインポート・デプロイされているリソースアダプタをバージョンアップします。	-
cjsetup	J2EE サーバのセットアップとアンセットアップ	J2EE サーバの環境をセットアップして、デフォルトのユーザ定義を作成します。	-
cjsleep	J2EE サーバの起動間隔	指定された秒数だけ実行を一時停止します。バッチファイルなどでサーバの起動間隔を空けるために使用します。このコマンドは、Windows 用です。	-
cjstartrecover	J2EE サーバのトランザクション回復	J2EE サーバのトランザクションを回復します。	-
cjstartsv	J2EE サーバの開始	J2EE サーバを開始します。	-
cjstopsv	J2EE サーバの停止	J2EE サーバを停止します。	更新

(凡例)

更新：更新系コマンド。J2EE サーバの内容を更新、または構成情報を取得するコマンドです。

-：該当なし

注

-f オプションを指定した場合は、- (該当なし) になります。

サーバ管理コマンドの系統ごとの排他制御については、マニュアル「Cosminexus アプリケーションサーバアプリケーション設定操作ガイド」を参照してください。

セットアップ系コマンド（cjenvsetup、cjenvupdate、cjrupdate、cjsetup）は単独で実行してください。ほかのセットアップ系コマンドと同時に実行しないでください。

cjdumpsv (J2EE サーバのスレッドダンプの取得)

形式

cjdumpsv [<サーバ名称>]

機能

Web ブラウザや EJB クライアントに応答が返らない場合、J2EE サーバで障害が発生した場合、アプリケーションの挙動を調査したい場合などに、JavaVM のスレッドダンプを取得します。

コマンドを実行すると、サーバ標準出力ログ（cjstdout.log）および次のファイルに、スレッドダンプが出力されます。

- Windows の場合
 <J2EE サーバの作業ディレクトリ>%ejb%<サーバ名称>%javacore*.txt
- UNIX の場合
 <J2EE サーバの作業ディレクトリ>/ejb/<サーバ名称>/javacore*.txt

なお、<J2EE サーバの作業ディレクトリ>とは、usrconf.cfg（J2EE サーバ用オプション定義ファイル）の ejb.public.directory キーに指定したディレクトリです。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

戻り値

- 0 :
 正常終了しました。
- 1 :
 異常終了しました。

注意事項

UNIX の場合、このコマンドの実行には、root 権限、または Component Container 管理者の権限が必要です。

コマンドの引数にサーバ名称を指定する場合、`cjsetup` コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjenvsetup (Component Container 管理者のセットアップ)

形式

`cjenvsetup`

機能

インストールファイルの属性（オーナー，グループ，および権限）を変更します。なお，このコマンドは UNIX 用です。

インストールディレクトリ（`/opt/Cosminexus/CC`）に設定されている属性（オーナー，グループ）が，Component Container 管理者および Component Container 管理者グループとして設定されます。

引数

なし

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

このコマンドを実行する前に，インストールディレクトリ（`/opt/Cosminexus/CC`）をシンボリックリンクしている場合は，リンク先ディレクトリの属性（オーナー，グループ）を Component Container 管理者および Component Container 管理者グループに変更する必要があります。

このコマンド実行時に，次の操作をしないでください。操作した場合の動作は保証されません。

- インストールディレクトリ（`/opt/Cosminexus/CC`）配下のファイル操作
- サーバの起動・停止などの操作

インストールディレクトリ配下でシンボリックリンクを使用している場合，リンクが指しているファイルまたはディレクトリが存在しない状態でこのコマンドを実行しないでください。必ずリンク先が存在することを確認してからコマンドを実行してください。

cjenvupdate (作業ディレクトリおよびユーザ定義ファイルの移行)

形式

```
cjenvupdate [-f]
```

機能

旧バージョンの Cosminexus Component Container で使用していた J2EE サーバが使用しているすべての旧形式の作業ディレクトリおよび次のユーザ定義ファイルを、本バージョンで使用する形式に移行します。

- usrconf.properties (J2EE サーバ用ユーザプロパティファイル)
- usrconf.cfg (J2EE サーバ用オプション定義ファイル)
- server.policy (J2EE サーバ用セキュリティポリシーファイル)
- usrconf.properties (サーバ管理コマンド用システムプロパティファイル)
- usrconf.bat (サーバ管理コマンド用オプション定義ファイル)
- usrconf.properties (Web コンテナサーバ用ユーザプロパティファイル)
- usrconf.cfg (Web コンテナサーバ用オプション定義ファイル)
- web.policy (SecurityManager 定義ファイル)

注 UNIX の場合は、usrconf です。

旧バージョンからのアップグレードインストール終了後、続けてこのコマンドを実行する必要があります。また、作業ディレクトリを移行するには、作業ディレクトリのあるドライブに、作業ディレクトリのディスク使用量の約 2 倍分の空き容量が必要です。

なお、<作業ディレクトリ>とは、usrconf.cfg (J2EE サーバ用オプション定義ファイル) の `ejb.public.directory` キーに指定したディレクトリです。

このコマンドを実行すると、移行対象の定義ファイルのバックアップが、定義ファイルと同じディレクトリに作成されます。作成されるバックアップファイルの名称は、「<移行対象定義ファイル名>.envupdate_bak」になります。バックアップが作成されるときに同名のファイルがすでにあった場合、バックアップファイル名は「<移行対象定義ファイル名>.envupdate_bak_?」(? は 1 ~ 2147483647 の連番)となります。なお、バックアップファイルは、このコマンドの完了後、自動で削除されません。

アプリケーションサーバの移行の手順については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

引数

-f

移行できない DD ファイルがある場合でも強制的に移行します。

移行後の作業

このコマンドによって作業ディレクトリが移行される場合、移行が完了すると、各 J2EE サーバの作業ディレクトリの直下にバックアップ作業ディレクトリが作成されます¹。作業ディレクトリには移行後の情報（ファイル）、バックアップ作業ディレクトリには移行前の情報（ファイル）が保存されています。なお、作業ディレクトリが「・・・¥<ディレクトリ名>」または「・・・/<ディレクトリ名>」の場合、バックアップ作業ディレクトリは、「・・・¥<ディレクトリ名>¥<ディレクトリ名>_old¥」または「・・・/<ディレクトリ名>¥<ディレクトリ名>_old/」²になります。移行後、J2EE サーバの起動確認、J2EE アプリケーションの動作確認、および J2EE サーバの停止確認が終わり、移行前の作業ディレクトリの情報（ファイル）が不要な場合、バックアップ作業ディレクトリを削除してください。ただし、バックアップ作業ディレクトリは、別の場所に保存しておくことを推奨します。

注 1

作業ディレクトリが移行されない場合、バックアップ作業ディレクトリは作成されません。

注 2

すでにバックアップ作業ディレクトリが存在している場合、「・・・¥<ディレクトリ名>¥<ディレクトリ名>_old¥」または「・・・/<ディレクトリ名>¥<ディレクトリ名>_old/」は、「・・・¥<ディレクトリ名>¥<ディレクトリ名>_old?¥」または「・・・/<ディレクトリ名>¥<ディレクトリ名>_old?/」(? は、1 ~ 2147483647 の連番) にリネームされます。

エラー時の対処方法

作業ディレクトリ移行コマンドの実行に失敗した場合、コマンドが出力するメッセージから障害が発生したファイルパスを特定できます。メッセージの例を次に示します。

```
KDJE30220-I Migration of the definition files of the old version is in progress.
KDJE30221-I The definition files of the old version have been migrated
successfully.
KDJE30204-I The server checking is in progress. Server name = [サーバ名称]
KDJE30202-E The Deployment Descriptor cannot be converted. File = [ファイルパス],
Reason = [詳細情報]
KDJE30205-E The server will not be migrated because a problem occurred in the
Deployment Descriptor. Server name = [サーバ名称]
```

詳細情報には、例外情報などが出力されます。この詳細情報を参照して問題を取り除き、ディスクに空き容量が十分であることを確認して、再度コマンドを実行してください。

詳細情報から原因が判断できない場合、または見直し後再実行してもエラーが発生した場合、保守員に連絡してください。

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。

注意事項

UNIX の場合、このコマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

06-50 以前のバージョンのアプリケーションサーバからアップグレードインストールしたとき、スタンダードモードまたはアドバンスドモードの J2EE サーバ上に JdbcDbpsv.jar ファイルと JdbcDbpsvEX.jar ファイルで構成されたデータソースが存在する場合、DB Connector (DBConnector_DABJ_XA.rar) の J2EE リソースアダプタへ置き換えます。また、データソースの表示名および属性情報は J2EE リソースアダプタへ引き継がれます。表示名が同じ J2EE リソースアダプタがすでに存在した場合は、表示名の最後に 1 ~ 2147483647 の連番を付けて一意な表示名に変更します。ただし、J2EE リソースのリソースアダプタ (インポートされたリソースアダプタ) は作成されません。データソースの拡張 DD ファイル (hitachi-ds.xml) がスクランブルされている場合、DB Connector の Java EE 仕様に準拠した DD ファイル (ra.xml) および拡張 DD ファイル (hitachi-ra.xml) はスクランブルされます。

J2EE リソースのリソースアダプタおよび J2EE リソースアダプタの拡張 DD ファイル (hitachi-ra.xml) がスクランブルされている場合、スクランブルされていない Java EE 仕様に準拠した DD ファイル (ra.xml) はスクランブルされます。

cjgetsysinfo (OS 状態情報の取得)

形式

```
cjgetsysinfo [-f <OS状態情報出力ファイルパス>]
```

機能

OS の状態情報を取得します。OS の状態情報取得コマンドの実行結果を、標準出力または -f オプションで指定したファイルへ出力します。

Windows の場合の OS の状態情報取得コマンドを次に示します。

```
netstat -e
netstat -s
netstat -an
set
```

2. J2EE サーバで使用するコマンド

注 08-00 以前のバージョンでは、netstat -a コマンドが使用されます。

UNIX の場合の OS の状態情報取得コマンドを次の表に示します。

表 2-3 OS の状態情報取得コマンドの一覧

AIX	HP-UX	Linux	Solaris
df -k	df -kt	df	df -k
ps -elf	top -s 1 -d 1	ps -eflm	ps -efll
ps -A -m -o THREAD	ps -efl	vmstat	vmstat
vmstat -t 1 1	vmstat	netstat -s	vmstat -s
vmstat -s	vmstat -s	netstat -an ¹	netstat -i
lsps -s	netstat -i	iostat ²	netstat -m
netstat -i	netstat -an ¹	top -b -n 1	netstat -an ¹
netstat -m	iostat	sysctl -a	iostat
netstat -an ¹	sar -A 1	sar -A 1 ²	mpstat
iostats	swapinfo -a	rpm -qa	swap -l
svmon -P	kmtune	rpm -qai	sar -A 1
svmon -G	swlist	uname -a	sysdef
sar -A 1	swlist -l product	env	pkginfo
instfix -i	uname -a	set	patchadd -p
lslpp -hac	env	ipcs	uname -a
uname -a	set	ipcs -t	env
env	ipcs -a	ipcs -p	set
set		ipcs -c	
ipcs -a		ipcs -u	
		ipcs -l	

注 1

08-00 以前のバージョンでは、netstat -a コマンドが使用されます。

注 2

sysstat パッケージがインストールされている必要があります。

引数

-f <OS 状態情報出力ファイルパス >

OS の状態情報を出力するファイルパスを指定します。

出力例

AIX の場合の例を次に示します。

```

cjgetsysinfo start time:20040804180232 ← 情報取得開始時間
----- df -k
Filesystem      1024-blocks    Free %Used    lused %lused Mounted on
/dev/hd4         32768          12252  63%      1982   13% /
:
/dev/Cosminexus  4194304        1367424  68%      20418   2% /opt/Cosminexus
----- ps -elf
  F S      UID  PID  PPID  C  PRI  NI ADDR  SZ  WCHAN  STIME  TTY  TIME CMD
200003 A    root    1    0    0  60  20 28034 1876          Jul 15  -  0:31 /etc/init
:
:
cjgetsysinfo end time:20040804180316 ← 情報取得終了時間

```

右側の注釈: } コマンド および 実行結果

戻り値

0 :

正常終了しました。

1 以上 :

異常終了しました。

注意事項

指定したパスが存在しない場合、エラーとなります。

環境によって、OS の状態情報取得コマンドが見つからない場合、そのコマンドの実行結果は出力されません。

このコマンドに掛かる処理時間は、OS の状態情報取得コマンドの処理時間に依存します。

cjrarupdate (リソースアダプタのバージョンアップ)

形式

リソースアダプタをバージョンアップする場合

```

cjrarupdate -type {dbconnector|rar} [-force] [-f <ファイルパス>
...]
[-backupto <バックアップ格納先ディレクトリ>]

```

作業ディレクトリ環境を回復する場合

```

cjrarupdate -recoverfrom <バックアップRARディレクトリ>

```

機能

リソースアダプタをバージョンアップする場合

Cosminexus Component Container にインポート・デプロイされているリソースアダプタをバージョンアップします。

リソースアダプタの移行には、作業ディレクトリのディスク使用量の約 2 倍以上の空き容量が必要です。また、アップグレードインストールの場合は、移行コマン

2. J2EE サーバで使用するコマンド

ド `ejenvupdate` によって移行処理が問題なく完了していることを確認してから、このコマンドを実行してください。

なお、<作業ディレクトリ>とは、`usrconf.cfg` (J2EE サーバ用オプション定義ファイル) の `ejb.public.directory` キーに指定したディレクトリです。

注

移行するリソースアダプタ数やコマンド実行回数によって、必要なディスクの空き容量は変わります。移行に必要なディスク容量については、アプリケーションサーバの移行の手順で確認してください。アプリケーションサーバの移行の手順については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

作業ディレクトリ環境を回復する場合

Cosminexus Component Container にインポート・デプロイされているリソースアダプタのバージョンアップに失敗した場合、作業ディレクトリ環境を回復します。

引数

`-type {dbconnector|rar}`

バージョンアップするリソースアダプタの種類を指定します。

- `dbconnector`

DB Connector の場合に指定します。

- `rar`

Cosminexus Component Container がサポートしている DB Connector 以外のリソースアダプタの場合に指定します。

`-force`

強制的にバージョンアップを行います。この引数が指定された場合、リソースアダプタのバージョンにかかわらずリソースアダプタをバージョンアップします。

`-f <ファイルパス>`

- 移行対象として DB Connector 以外のリソースアダプタを指定した場合 (`-type rar` を指定した場合)

旧バージョンの既存リソースアダプタと置き換えるリソースアダプタのファイルパスを指定します。ファイルパスは、複数回指定できますが、`-f` オプションは複数回指定できません。

`-f` オプションに Connector 1.5 の仕様に準拠するリソースアダプタを指定した場合、メッセージ `KDJE42356-E` が出力されます。

- 移行対象として DB Connector を指定した場合 (`-type dbconnector` を指定した場合)

この引数は無視されます。

`-backupto <バックアップ格納先ディレクトリ>`

作業ディレクトリのバックアップの格納先を指定します。このオプションを省略し

た場合、<作業ディレクトリ>下にバックアップディレクトリが作成されます。

作業ディレクトリのあるディスクの容量に余裕がない場合、またはファイルシステムのパス長制限でバックアップが作成できない場合に、このオプションを指定してください。なお、このオプションでバックアップ格納先ディレクトリを指定する場合、次のことに注意してください。

- 複数回このコマンドを実行する場合には、同一ディレクトリを指定してください。
- 長いパス名を指定すると、バックアップ作成に失敗することがあります。その場合、失敗したバックアップを削除してから、短いパス名を指定して再度実行してください。
- Cosminexus Component Container のインストールディレクトリを含まないようにしてください。

-recoverfrom <バックアップ RAR ディレクトリ>

回復元となる作業ディレクトリのバックアップ RAR のディレクトリを指定します。

移行後の作業

移行が完了した場合、各 J2EE サーバの作業ディレクトリ、または -backupto オプションで指定したディレクトリの直下にバックアップ作業ディレクトリが作成され、作業ディレクトリには移行後の情報（ファイル）、バックアップ作業ディレクトリには移行前の情報（ファイル）が保存されています。なお、作業ディレクトリが「・・・¥<ディレクトリ名>¥」または「・・・/<ディレクトリ名>/」の場合、バックアップ作業ディレクトリは、「・・・¥<ディレクトリ名>¥rarupdate_bk¥」または「・・・/<ディレクトリ名>/rarupdate_bk/」になります。移行後、リソースアダプタの開始、停止およびデータベースへの接続確認が終わり、移行前の作業ディレクトリの情報（ファイル）が不要な場合、バックアップ作業ディレクトリを削除してください。ただし、バックアップ作業ディレクトリは、別の場所に保存しておくことを推奨します。

注

バックアップの対象ディレクトリは、リソースアダプタに関連するディレクトリです。

すでにバックアップ作業ディレクトリが存在している場合、「・・・¥<ディレクトリ名>¥rarupdate_bk¥」または「・・・/<ディレクトリ名>/rarupdate_bk/」は、「・・・¥<ディレクトリ名>¥rarupdate_bk_?¥」または「・・・/<ディレクトリ名>/rarupdate_bk_?/」(? は、コマンド実行時のミリ秒までの日時) にリネームされます。

バージョンタグの比較方法

移行する RAR ファイルによっては、バージョンタグを比較できないため、移行処理が正常に終了しないことがあります。次の手順でバージョンタグを比較してください。

1. バージョン文字列を、RAR ファイルに含まれる ra.xml または J2EE サーバにインポート、デプロイされたリソースアダプタの ra.xml の <version> タグから取得します。

2. J2EE サーバで使用するコマンド

2. バージョン文字列が、次の形式で指定されている場合は、比較可能な形式に編集します。

< 連続した数字 1>.< 連続した数字 2>< 数字以外から開始する文字列 >

比較可能な形式

$V_1V_2-R_1R_2-S_1S_2$ (または $V_1V_2-R_1R_2$)

V_1, V_2, R_1 および R_2 : 0~9の数値

S_1 : /

S_2 : A~Z (英字大文字)

編集手順は次のとおりです。

- < 連続した数字 1> と < 連続した数字 2> の間に半角でハイフン (-) を追加します。
< 連続した数字 1>-< 連続した数字 2>< 数字以外から開始する文字列 >
- < 連続した数字 1> が 10 未満の場合、先頭に 0 を付加します。
- < 連続した数字 2> が 10 未満の場合、末尾に 0 を付加します。

編集例

1.0	01-00
1.1	01-10
1.12	01-12
2.0	02-00

バージョン文字列が、 $V_1V_2-R_1R_2[-S_1S_2]$ の形式で指定されている場合、次の順序でバージョンの大小比較を行います。

- $V_1V_2-R_1R_2$ 部を文字列として比較して、Unicode の辞書順で大きい方を新バージョンとします。(例: 01-00-/Z と 02-00 では、02-00 の方が新しい)
- $V_1V_2-R_1R_2$ 部が同一の場合、 $-S_1S_2$ 部が存在している方を新バージョンとします。(例: 02-00-/A と 02-00 では、02-00-/A の方が新しい)
- $V_1V_2-R_1R_2$ 部が同一の場合でどちらも $-S_1S_2$ 部が存在しているとき、次のようになります。
 1. S_2 を文字列として比較し、Unicode の辞書順で大きい方を新バージョンとします。
 2. S_1 を文字列として比較し、/ > P > S の順で新バージョンとします。

(例: 02-00-/A と 02-00-/B では、02-00-/B の方が新しい)

02-01-SB と 02-01-/A では、02-01-SB の方が新しい

02-01-SA と 02-01-PA では、02-01-PA の方が新しい

02-01-PA と 02-01-/A では、02-01-/A の方が新しい)

- どちらかまたは両方の形式が異なる場合、バージョン文字列全体を文字列比較し、Unicode の辞書順で大きい方を新バージョンとします。

なお、ra.xml の <version> タグの値がサポート外の文字列を使用している場合、正しい比較ができません。次の対処が必要です。

- 入れ替える RAR ファイルが新しいバージョンでも古いバージョンと認識されてしま

う場合は、-force オプションを指定して実行してください。

- 誤って古いバージョンの RAR ファイルで J2EE サーバにインポートまたはデプロイされたリソースアダプタを移行してしまった場合、バックアップ作業ディレクトリから元の環境を復旧してください。バックアップ作業ディレクトリについては、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

入力例

-f オプションに、ファイルパスを複数回指定する場合

```
cjrrarupdate -type rar -f file1 file2 file3
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

Windows の場合、このコマンドの実行には、Administrator 権限が必要です。

UNIX の場合、このコマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

-type オプションはコマンド名の直後に指定し、dbconnector または rar を指定する必要があります。-force オプション、-f オプションおよび -backupto オプションの順序は任意です。

このコマンドは同時に実行しないでください。

J2EE サーバが起動している場合、または J2EE サーバの起動チェック中に問題が発生した場合は、メッセージを出力して処理を中断します。

次の RAR ファイルは、バージョンアップの対象外です。

- RAR ファイルの内容を変更している場合。
- Cosminexus Component Container が提供、またはサポートしている RAR ファイル名と同名のファイルを作成してインポートしている場合。
- Cosminexus Component Container が提供、またはサポートしている RAR ファイル名と、数字および文字を組み合わせて生成されたファイル名の場合。
- Connector 1.5 の仕様に準拠するリソースアダプタの場合。

Connector 1.5 の仕様に準拠するリソースアダプタをバージョンアップする場合、再インポートおよび定義を再設定してください。

2. J2EE サーバで使用するコマンド

TP1/Message Queue-Access の移行をする場合に、新しい RAR ファイルが同一バージョンと認識されることがあります。その場合は `-force` オプションを指定して強制的に移行処理を実行してください。

Cosminexus Component Container 以外が提供するリソースアダプタ (uCosminexus TP1 Connector および TP1/Message Queue-Access) は、バージョンアップした場合にユーザがプロパティを追加する必要があることがあります。その場合、このコマンドで移行しただけでは動作しません。リソースアダプタのマニュアルに従って設定をしてください。

Cosminexus Component Container 以外が提供するリソースアダプタ (uCosminexus TP1 Connector および TP1/Message Queue-Access) は、バージョンアップに失敗する場合があります。その場合は、バージョンアップに失敗したリソースアダプタを削除し、再度インポートしたあとプロパティを設定してください。なお、削除する前に設定を控えておくことを推奨します。

入れ替える RAR ファイルが新しいバージョンでも同一バージョンまたは古いバージョンと認識されてしまう場合は、`-force` オプションを指定して再度このコマンドを実行してください。バージョンに関係なく強制的に移行されます。

<Cosminexus インストールディレクトリ>¥CC¥DBConnector¥ (Windows の場合)、または /opt/Cosminexus/CC/DBConnector/ (UNIX の場合) 以下にある rar ファイルを変更、削除した場合、移行処理が正常に動作しないことがあるため、変更、削除しないでください。

バックアップ中に例外が発生した場合は、例外メッセージ出力後、このコマンドは終了しないで処理を続けます。このときバックアップディレクトリは作成されません。

リソースアダプタが存在しない J2EE サーバについては、バックアップを作成しないで処理を続けます。

バックアップを作成していない J2EE サーバについては、移行処理をしません。

DB Connector の Connector 属性ファイルの <config-property> タグに、`loginTimeout`、`PreparedStatementPoolSize`、`CallableStatementPoolSize`、または `CancelStatement` プロパティが存在しない場合にこのコマンドで移行をすると、移行前に存在しなかったプロパティの <config-property-value> は、移行後に次の値になります。

- `loginTimeout:0`
- `PreparedStatementPoolSize:0`
- `CallableStatementPoolSize:0`
- `CancelStatement:false`

cjsetup (J2EE サーバのセットアップとアンセットアップ)

形式

セットアップの場合

```
cjsetup [<サーバ名称>]
```

アンセットアップの場合

```
cjsetup -d <サーバ名称>
```

機能

指定したサーバ名称の J2EE サーバの環境をセットアップします。

次のディレクトリにセットアップした J2EE サーバ用のデフォルトのユーザ定義を作成します。

- Windows の場合

```
<Cosminexus のインストールディレクトリ >%CC%\server\usrconf\ejb< サーバ名称 >%
```

- UNIX の場合

```
/opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称 >/
```

なお、-d オプションを指定した場合、次に示す J2EE サーバの環境 (ディレクトリおよびファイル) を削除します。

ユーザ定義ファイルの格納用ディレクトリ以下

- Windows の場合

```
<Cosminexus のインストールディレクトリ >%CC%\server\usrconf\ejb< サーバ名称 >%
```

- UNIX の場合

```
/opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称 >/
```

一時作業用ディレクトリ以下

- Windows の場合

```
<Cosminexus のインストールディレクトリ >%CC%\server\repository< サーバ名称 >%
```

- UNIX の場合

```
/opt/Cosminexus/CC/server/repository/< サーバ名称 >/
```

作業用ディレクトリ以下 (デフォルト)

- Windows の場合

```
<Cosminexus のインストールディレクトリ >%CC%\server\public\ejb< サーバ名称 >%
```

- UNIX の場合

```
/opt/Cosminexus/CC/server/public/ejb/< サーバ名称 >/
```

2. J2EE サーバで使用するコマンド

Web アプリケーションの格納用ディレクトリ以下 (デフォルト)

- Windows の場合
<Cosminexus のインストールディレクトリ>¥CC¥server¥public¥web¥< サーバ名称 >¥
- UNIX の場合
/opt/Cosminexus/CC/server/public/web/< サーバ名称 >/

なお、作業用ディレクトリ、および Web アプリケーションの格納用ディレクトリについては、ユーザ定義ファイル (usrconf.cfg) の `ejb.public.directory` キーにデフォルト値以外が指定されていた場合、そのディレクトリ以下も削除します。

引数

< サーバ名称 >

セットアップ、またはアンセットアップする J2EE サーバのサーバ名称を指定します。

サーバ名称には、半角英数字 (0 ~ 9, A ~ Z, a ~ z), アンダースコア (_) またはハイフン (-) を使用できます。サーバ名称の先頭の文字は英数字でなければなりません。また、サーバ名称は、255 文字以下、かつマニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の J2EE サーバの作業ディレクトリの説明を参照して見積もった範囲内の長さで指定してください。

サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。ただし、アンセットアップ時は省略できません。また、ホスト名称にサーバ名称として使用できない文字が含まれている場合はエラーになります。

-d

J2EE サーバの環境を削除する場合に指定します。サーバ名称は省略できません。

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

UNIX の場合、このコマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

UNIX の場合にアンセットアップ (`cjsetup -d <サーバ名称>`) を実行すると、削除対象のディレクトリ、ファイルのアクセス権の設定および使用状態に関係なく、すべて削除します。

セットアップ済みのサーバ名称を指定して再セットアップ (`cjsetup <サーバ名称>`) を行うと、エラーとなります。

アンセットアップを実行するときにコマンドの引数に指定するサーバ名称には、セットアップで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。ただし、Windows の場合、サーバ名称の大文字・小文字は区別されません。すでに作成済みのサーバ名称と、大文字・小文字の区別以外は一致している文字列を `cjsetup` のサーバ名称として指定した場合、次のように動作します。

- セットアップの場合 (`cjsetup <サーバ名称>`)
セットアップに失敗します。
- アンセットアップの場合 (`cjsetup -d <サーバ名称>`)
すでに作成済みである、`<サーバ名称>` と大文字・小文字の区別以外は一致しているサーバが削除されます。

サーバ管理コマンド、J2EE サーバの開始または停止コマンドを実行する場合、これらのコマンドにサーバ名称を引数として指定するときは、このコマンドで作成したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

起動中の J2EE サーバに対して、アンセットアップ (`cjsetup -d <サーバ名称>`) を実行しないでください。

Windows の場合、削除対象のファイルの属性が読み取り専用、または使用中のときに、アンセットアップ (`cjsetup -d <サーバ名称>`) を実行しないでください。

cjsleep (J2EE サーバの起動間隔)

形式

`cjsleep` [`<スリープ時間>`]

機能

指定された秒数だけ実行を一時停止します。バッチファイルなどでサーバの起動間隔を空けるために使用します。なお、このコマンドは、Windows 用です。

スリープ時間に指定した値が 1 ~ 2147483647 の正の整数でない場合は、デフォルト値として 10 が使用されます。

引数

`<スリープ時間>`

J2EE サーバの起動を遅らせる時間を、1 ~ 2147483647 (単位：秒) の正の整数で指定します。

戻り値

0 :

正常終了しました。

0 以外：

異常終了しました。

cjstartrecover (J2EE サーバのトランザクション回復)

形式

```
cjstartrecover [<サーバ名称>] [-nosecurity] [-nostartapp]
                [[-p <キー名称>=<値>] ...] [-t <タイム監視時間>]
```

機能

J2EE サーバをリカバリ処理専用のリカバリモードとして起動します。

リカバリモードで起動した J2EE サーバは、トランザクションのリカバリ処理を完了したあと、J2EE サーバを停止します。

引数

<サーバ名称>

リカバリする J2EE サーバのサーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nosecurity

SecurityManager を解除する場合に指定します。リソースアクセス時の権限チェックのオーバーヘッドを削減できますが、J2EE アプリケーションは任意のリソースへのアクセスができるようになるため、セキュリティが低下します。

-nostartapp

無限ループなどの動作不正の J2EE アプリケーションがある場合に、J2EE サーバを強制終了してもサーバ再起動時に J2EE アプリケーションが自動開始されてしまい、アプリケーションの入れ替えができない状態を回避するときなどに指定します。このオプションを指定して J2EE サーバを起動し、登録済みの J2EE アプリケーションに対して次の操作をした場合、次回このオプションを省略して J2EE サーバを起動すると、その J2EE アプリケーションに対する開始処理をしません。

- アプリケーションの属性設定
- リソース (フィルタ, WAR, EJB-JAR, またはリソースアダプタ) の追加
- リソース (フィルタ, WAR, EJB-JAR, またはリソースアダプタ) の削除
- ライブラリ JAR のインポート
- ライブラリ JAR の削除
- アプリケーションの入れ替え
- CMP2.x Entity Bean 用 SQL 文の生成

なお、開始状態の J2EE アプリケーションを実行時情報を含めてエクスポートした

場合、この EAR ファイルをインポートすると、`-nostartapp` オプションの指定に関係なく J2EE アプリケーションは自動で開始されます。

`-p <キー名称>=<値>`

J2EE サーバを実行する JavaVM のシステムプロパティを指定します。指定できるのは、`usrconf.properties` (J2EE サーバ用ユーザプロパティファイル) のキーです。`usrconf.properties` (J2EE サーバ用ユーザプロパティファイル) については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」を参照してください。

値にスペースや記号 (" | ", " & " など) が含まれる文字列を指定する場合は、値をダブルクォーテーション (" ") で囲んでください。

指定したキー名称が、J2EE サーバ用ユーザプロパティファイルに定義されているプロパティのキー名称と重複した場合、`-p` オプションで指定したプロパティ値が優先されます。指定したプロパティの値に誤りがあった場合、指定したプロパティキー名称の仕様に従います。指定したプロパティのキー名称に誤りがあった場合、その指定を無視して処理を続けます。

`-t <タイマ監視時間>`

タイマ監視時間を 0 ~ 3600 の整数 (単位: 秒) で指定します。

指定したタイマ監視時間内にリカバリ処理が終了しない場合、J2EE サーバは強制停止します。このオプションを省略した場合、0 秒が設定されます。

タイマ監視時間に 0 を指定した場合、タイマ監視を行わないで、J2EE サーバが終了するまで待ち続けます。

入力例

```
cjstartrecover MyServer -t 300
```

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。
- 2 :
タイムアウトにより強制停止しました。

注意事項

Windows の場合、このコマンドの実行には、Administrator 権限が必要です。

UNIX の場合、このコマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

2. J2EE サーバで使用するコマンド

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：< タイマ監視時間 > -t）やオプション名と値を非対応にすること（例：-t < キー名称 > = < 値 > -p < タイマ監視時間 >）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

コマンドの引数に -t オプションを指定する場合は、そのほかの引数の後ろにする必要があります。また、タイマ監視時間には、トランザクションタイムアウト時間と J2EE サーバの開始処理に掛かる時間を考慮して長めに設定しておく必要があります。

コマンドの引数に -p オプションを複数指定する場合、コマンドラインに指定できる文字長は使用する OS によって制限があります。

cjstartsv (J2EE サーバの開始)

形式

```
cjstartsv [<サーバ名称>] [-nosecurity] [-nostartapp]
```

機能

指定したサーバ名称の J2EE サーバを開始します。サーバ名称を省略したときは、ホスト名称と同じサーバ名称が設定された J2EE サーバを開始します。

引数

<サーバ名称>

開始する J2EE サーバのサーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nosecurity

SecurityManager を解除する場合に指定します。リソースアクセス時の権限チェックのオーバーヘッドを削減できますが、J2EE アプリケーションは任意のリソースへのアクセスができるようになるため、セキュリティが低下します。

-nostartapp

無限ループなどの動作不正の J2EE アプリケーションがある場合に、J2EE サーバを強制終了してもサーバ再起動時に J2EE アプリケーションが自動開始されてしまい、アプリケーションの入れ替えができない状態を回避するときなどに指定します。このオプションを指定して J2EE サーバを起動し、登録済みの J2EE アプリケーションに対して次に示す操作をした場合、次回このオプションを省略して J2EE サーバを起動すると、その J2EE アプリケーションに対する開始処理をしません。

- アプリケーションの属性設定

- リソース (フィルタ, WAR, EJB-JAR, またはリソースアダプタ) の追加
- リソース (フィルタ, WAR, EJB-JAR, またはリソースアダプタ) の削除
- ライブラリ JAR のインポート
- ライブラリ JAR の削除
- アプリケーションの入れ替え
- CMP2.x Entity Bean 用 SQL 文の生成

なお、開始状態の J2EE アプリケーションを実行時情報を含めてエクスポートした場合、この EAR ファイルをインポートすると、`-nostartapp` オプションの指定に関係なく J2EE アプリケーションは自動開始されます。

ユーザ定義ファイル

Windows の場合

- `<Cosminexus のインストールディレクトリ>%CC%\server\usrconf\ejb\<サーバ名称>\usrconf.cfg`
- `<Cosminexus のインストールディレクトリ>%CC%\server\usrconf\ejb\<サーバ名称>\usrconf.properties`

UNIX の場合

- `/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/usrconf.cfg`
- `/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/usrconf.properties`

戻り値

0 :

正常終了しました。

1 以上 (143 以外) :

異常終了しました。

143 :

コマンド実行中に次の操作をしました (Windows の場合)。

- コマンドを起動しているコマンドプロンプトのウィンドウを「x」ボタンを押下して閉じた。
- Windows システムをシャットダウンした。
- Windows システムをログオフした。

注意事項

J2EE サーバを自動起動するように設定できます。J2EE サーバの起動については、マニュアル「Cosminexus アプリケーションサーバ システム構築・運用ガイド」を参照してください。

Windows の場合、このコマンドの実行には、Administrator 権限が必要です。

2. J2EE サーバで使用するコマンド

UNIX の場合、このコマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjstopsv (J2EE サーバの停止)

形式

```
cjstopsv [<サーバ名称>]
          [-f]-fd [<スレッドダンプ取得間隔> <スレッドダンプ取得回数>]
          [-wait <終了待ち時間>]
```

機能

サーバ名称の J2EE サーバを停止します。サーバ名称を省略したときは、ホスト名称がサーバ名称になります。

引数

<サーバ名称>

停止する J2EE サーバのサーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-f

J2EE サーバを強制停止します。

Management Server から論理サーバとして J2EE サーバを起動している場合、コンソールからのこのコマンドによる強制停止はできません。強制停止する場合は、Management Server から実行します。論理サーバの停止については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」を参照してください。

-fd

J2EE サーバの障害情報を取得したあと、J2EE サーバを強制停止します。

<スレッドダンプ取得間隔>

スレッドダンプを取得する間隔を 1 ~ 60 の整数 (単位: 秒) で指定します。省略した場合の初期値は 3 秒です。

<スレッドダンプ取得回数>

スレッドダンプを取得する回数を 1 ~ 60 の整数 (単位: 回) で指定します。省略した場合の初期値は 10 回です。

取得する障害情報を次に示します。

1. OS の稼働状況

Windows の場合の出力先 :

```
<J2EE サーバの作業ディレクトリ >¥ejb¥< サーバ名称 >¥cjsysinfo.< 出力時間 >.txt
```

UNIX の場合の出力先 :

```
<J2EE サーバの作業ディレクトリ >/ejb/< サーバ名称 >/cjsysinfo.< 出力時間 >.txt
```

< 出力時間 > は , yymmddHHMMSS の形式になります。

なお , 出力内容については , 「cjgetsysinfo (OS 状態情報の取得)」を参照してください。

2. JavaVM のスレッドダンプ

< スレッドダンプ取得間隔 > , < スレッドダンプ取得回数 > で指定した間隔と回数でスレッドダンプを取得します。J2EE サーバプロセスの状態によっては指定間隔 , 指定回数でスレッドダンプが出力されない場合もあります。なお , 出力内容については , 「cjdumpsv (J2EE サーバのスレッドダンプの取得)」を参照してください。

3. J2EE サーバのクラッシュダンプ (Windows の場合) または core ファイル (UNIX の場合)

最後のスレッドダンプ取得要求後 , スレッドダンプ取得間隔時間経過後にクラッシュダンプまたは core ファイルを出力します。

クラッシュダンプの出力先 :

Windows Server 2003 , Windows XP と Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista ではクラッシュダンプの出力先は異なります。

- ・ Windows Server 2003 , Windows XP の場合

```
< ワトソン博士のクラッシュダンプ出力先 > ¥user.dmp
```

注 : ワトソン博士のクラッシュダンプ作成オプションの設定が必要です。

- ・ Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合

```
< 環境変数 CJMEMDUMP_PATH で指定したディレクトリ >
```

```
¥cjmemdump.dmp
```

注 : 環境変数 CJMEMDUMP_PATH が指定されていないか , 指定したディレクトリが存在しない場合 , クラッシュダンプは出力されません。また , 指定したディレクトリにクラッシュダンプがすでに存在する場合 , クラッシュダンプは上書きされます。

core ファイルの出力先 :

```
<J2EE サーバの作業ディレクトリ >/ejb/< サーバ名称 >/core
```

注

Linux の場合 , core.< 強制停止された J2EE サーバのプロセス ID >

-wait < 終了待ち時間 >

終了待ち時間を 0 ~ 86400 の符号なし整数で指定します。デフォルト値は , 180 秒

2. J2EE サーバで使用するコマンド

です。

J2EE サーバへの正常終了要求を入力したあとから J2EE サーバの正常終了が完了するまでの `cjstopsv` の終了待ち時間を指定します。終了待ち時間に 0 を指定した場合、J2EE サーバが正常終了するまで無限に待ちます。このとき、`-f` または `-fd` オプションが指定されていても無視されます。

指定値に誤りがあった場合は、エラーを出力したあと、デフォルト値で正常終了を待ちます。

指定時間を超えても J2EE サーバが正常終了しなかった場合、次の表に示すほかの引数との組み合わせによる処理をしたあと、コマンドは終了コード 2 で異常終了します。

ほかの引数との組み合わせ	タイムアウト後の動作
<code>-f</code> および <code>-fd</code> が未指定の場合	メッセージを出力して終了します。このとき、J2EE サーバの停止はしません。
<code>-f</code> が指定されている場合	メッセージを出力し、J2EE サーバの強制停止をします。
<code>-fd</code> が指定されている場合	メッセージを出力し、障害情報とスレッドダンプを取得して J2EE サーバの強制停止をします。なお、障害情報取得完了まで <code>cjstopsv</code> は終了しません。

ユーザ定義ファイル

Windows の場合

- `<Cosminexus のインストールディレクトリ>\%CC%\server\usrconf\%ejb\<サーバ名称>\usrconf.cfg`
- `<Cosminexus のインストールディレクトリ>\%CC%\server\usrconf\%ejb\<サーバ名称>\usrconf.properties`

UNIX の場合

- `/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/usrconf.cfg`
- `/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/usrconf.properties`

戻り値

0 :

正常終了しました。

1 以上 :

異常終了しました。詳細については、次の表を参照してください。

終了コード	説明
2	<code>-wait</code> による正常終了監視タイムアウトです。
2 以外	パラメタ不正、内部エラーなどに関係するエラーの場合に出力されます（詳細は直前に表示されるメッセージを参照してください）。

注意事項

このコマンドで J2EE サーバを停止する場合、Enterprise Bean のビジネスメソッド、またはサーブレット / JSP のサービスメソッドが実行中のときは、該当するメソッドが終了するまで J2EE サーバは停止しません。

J2EE サーバを自動停止するように設定できます。J2EE サーバの停止については、マニュアル「Cosminexus アプリケーションサーバ システム構築・運用ガイド」を参照してください。

Windows の場合、このコマンドの実行には、Administrator 権限が必要です。

UNIX の場合、このコマンドの実行には、root 権限（Component Container 管理者を設定していない場合）、または Component Container 管理者の権限（Component Container 管理者を設定している場合）が必要です。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<終了待ち時間> -wait）やオプション名と値を非対応にすること（例：-f <スレッドダンプ取得回数> -fd <スレッドダンプ取得間隔>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

アプリケーションが存在する状態でこのコマンドを実行した場合、J2EE サーバを停止できないことがあります。この場合、-f オプションまたは -fd オプションを指定して、再度このコマンドを実行することで J2EE サーバを強制停止します。

Windows Server 2008 x86、Windows Server 2008 x64、Windows Server 2008 R2、Windows 7 または Windows Vista の場合で -fd オプションを指定する場合、環境変数 CJMEMDUMP_PATH は次のように設定してください。

- 環境変数 CJMEMDUMP_PATH で指定した保存先のディスクに十分な空き容量があることを確認してください。クラッシュダンプのファイルサイズは J2EE サーバの実メモリ所要量以上になります。
- 環境変数 CJMEMDUMP_PATH に日本語などのマルチバイト文字を含むディレクトリを指定しないでください。クラッシュダンプの出力に失敗する場合があります。
- クラッシュダンプの出力先ディレクトリには、存在するディレクトリを指定してください。

2.3 J2EE アプリケーションで使用するコマンド

ここでは、J2EE アプリケーションで使用するコマンドについて説明します。

J2EE アプリケーションで使用するコマンドの一覧を、次の表に示します。

表 2-4 J2EE アプリケーションで使用するコマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjaddapp	リソースの追加	インポート済みの EJB-JAR ファイル、WAR ファイル、RAR ファイル、またはフィルタを J2EE アプリケーションに追加します。	更新
cjchmodapp	アプリケーションのモード切り替え	J2EE アプリケーションの動作モードを変更します。	更新
cjdeleteapp ¹	J2EE アプリケーションの削除	J2EE アプリケーションまたは J2EE アプリケーションに含まれる EJB-JAR ファイル、WAR ファイル、RAR ファイル、またはフィルタを、指定された J2EE サーバから削除します。	更新
cjdeletelibjar	ライブラリ JAR の削除	ライブラリ JAR を J2EE アプリケーションから削除します。	更新
cjexportapp	J2EE アプリケーションのエクスポート	J2EE サーバ上の J2EE アプリケーションをエクスポートします。	更新
cjgenempsql	CMP2.x Entity Bean 用 SQL 文の生成	CMP2.x Entity Bean 用の SQL 文を生成します。	更新
cjgetappprop ¹	アプリケーションの属性の取得	J2EE アプリケーションまたは J2EE アプリケーションに含まれるリソースの属性を取得して、属性ファイルを生成します。	更新
cjgetstubsjar ¹	アプリケーションの RMI-IIOP スタブおよびインタフェースの取得	J2EE アプリケーションについて、RMI-IIOP スタブおよびインタフェースを取得します。	更新
cjimportapp ¹	J2EE アプリケーションのインポート	J2EE アプリケーションを J2EE サーバにインポートします。	更新
cjimportlibjar	ライブラリ JAR のインポート	J2EE アプリケーションにライブラリ JAR をインポートします。	更新

コマンド名称	分類	概要	コマンドの系統
cjlistapp ¹	アプリケーションの一覧表示	すべての J2EE アプリケーションについての名称と状態, J2EE アプリケーションに含まれる EJB-JAR ファイル, WAR ファイル, または RAR ファイルの一覧を標準出力に出力します。	参照
cjlistlibjar	ライブラリ JAR の一覧表示	J2EE アプリケーションに含まれるライブラリ JAR の一覧を標準出力に出力します。	参照
cjreloadapp	アプリケーションのリロード	J2EE アプリケーションを停止しないで構成するクラスおよび JSP を入れ替えます。	更新
cjrenameapp	アプリケーション名の変更	J2EE アプリケーションの名称を変更します。	更新
cjreplaceapp ¹	アプリケーションの入れ替え	J2EE アプリケーションを入れ替えます。	更新
cjsetappprop ¹	アプリケーションの属性設定	アプリケーション, EJB-JAR ファイルに含まれるリソース, WAR ファイルに含まれるリソース, または RAR ファイルの属性を, 指定されたアプリケーション属性ファイルの値に変更します。	更新
cjstartapp ¹	J2EE アプリケーションの開始	J2EE アプリケーションを開始して, クライアントからのリクエストを受け取ることができるようにします。	更新
cjstopapp ¹	J2EE アプリケーションの停止	J2EE アプリケーションを停止して, クライアントからのリクエストを受け取らないようにします。	更新 ²

(凡例)

更新: 更新系コマンド。J2EE サーバの内容を更新, または構成情報を取得するコマンドです。
 参照: 参照系コマンド。J2EE サーバの内容を更新しないで, 構成状態を表示するだけのコマンドです。

注 1

テストモードでも実行できます。

注 2

-cancel オプションを指定した場合, 特権系コマンドになります。

サーバ管理コマンドの系統ごとの排他制御の条件については, マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

cjaddapp (リソースの追加)

形式

EJB-JAR ファイル, WAR ファイル, または RAR ファイルの場合

```
cjaddapp [<サーバ名称>] [-nameserver <プロバイダURL>]  
-type {ejb|war|rar} -name <J2EEアプリケーション名>  
-resname <表示名> [-resname <表示名> ...]
```

フィルタの場合

```
cjaddapp [<サーバ名称>] [-nameserver <プロバイダURL>]  
-type filter -name <J2EEアプリケーション名>  
-warname <WARファイルの表示名>  
-c <属性ファイルパス> [-c <属性ファイルパス> ...]
```

機能

EJB-JAR ファイル, WAR ファイルまたは RAR ファイルの場合

J2EE アプリケーションが存在する場合, インポート済みの EJB-JAR ファイル, WAR ファイル, または RAR ファイルを J2EE アプリケーションに追加します。また, 一度に複数のファイルの追加もできます。その際, 指定したファイルの順に処理されます。

J2EE アプリケーションが存在しない場合, J2EE アプリケーションを新規作成して, インポート済みの EJB-JAR ファイル, WAR ファイル, または RAR ファイルを J2EE アプリケーションに追加します。

フィルタの場合

J2EE アプリケーションおよび WAR ファイルが存在する場合, 指定した WAR ファイルにフィルタを追加します。追加するフィルタは, -c オプションに指定するフィルタ属性ファイルに指定します。また, 一度に複数のフィルタの追加もできます。その際, 指定されたフィルタの順に処理されます。

J2EE アプリケーションまたは WAR ファイルが存在しない場合, エラーとなります。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは, ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

`-type {ejb|war|rar|filter}`

追加するリソースの種別を指定します。

- `ejb` : EJB-JAR ファイル
- `war` : WAR ファイル
- `rar` : RAR ファイル
- `filter` : フィルタ

`-name <J2EE アプリケーション名>`

J2EE アプリケーション名を指定します。J2EE アプリケーション名は、1 以上の文字列長である必要があります。

新規に J2EE アプリケーションを作る場合、名称には英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) を使用できます。

`-resname <表示名>`

追加対象となる EJB-JAR ファイル, WAR ファイル, または RAR ファイルの表示名を指定します。

`-warname <WAR ファイルの表示名>`

フィルタの追加先となる WAR ファイルの表示名を指定します。

`-c <属性ファイルパス>`

フィルタ属性ファイルの入力元パスを指定します。

入力例

EJB-JAR ファイルの場合

```
cjaddapp MyServer -type ejb -name Appl -resname accountJsp
```

WAR ファイルの場合

```
cjaddapp MyServer -type war -name Appl -resname account-war
```

RAR ファイルの場合

```
cjaddapp MyServer -type rar -name Appl -resname account-rar
```

フィルタの場合

```
cjaddapp MyServer -type filter -name Appl -warname account-war
-c FilterProp.xml
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2. J2EE サーバで使用するコマンド

2:

排他エラーによってコマンドが実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要がありません。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <データソース表示名> -resname) やオプション名と値を非対応にすること (例: -nameserver <データソース表示名> -resname <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一度に複数のファイルを追加した場合、追加できないファイルがあったときでも、すべてのファイルに対して追加を試みます。一つでも追加できないファイルがあった場合、終了コードは 1 (異常終了) になります。

すでにアプリケーションに追加されている EJB-JAR ファイルと同じ表示名を持つ EJB-JAR ファイルは、アプリケーションに追加できません。

すでにアプリケーションに追加されている WAR ファイルと同じ表示名を持つ WAR ファイルは、アプリケーションに追加できません。

すでに WAR ファイルに追加されているフィルタまたはサーブレットと同じ表示名を持つフィルタは、WAR ファイルに追加できません。

すでにアプリケーションに追加されている RAR ファイルと同じ表示名を持つ RAR ファイルは、アプリケーションに追加できません。

ネイティブライブラリを含む RAR ファイルは、アプリケーションに追加できません。

すでに J2EE リソースアダプタとしてデプロイされている RAR ファイルの表示名と同じ表示名を持つ RAR ファイルは、アプリケーションに追加できません。

-c オプションで指定するフィルタ属性ファイルは、<filter-name> および <filter-class> タグが指定されている必要があります。

EJB-JAR ファイル、WAR ファイル、および RAR ファイルは、展開ディレクトリ形式のアプリケーションに追加できません。

拡張子が ".jar" ではない EJB-JAR ファイル, 拡張子が ".war" ではない WAR ファイル, および拡張子が ".rar" ではない RAR ファイルは, application.xml を含まないアプリケーションに追加できません。

フィルタは, web.xml を含まない WAR ファイルに追加できません。

指定したアプリケーション名が 255 文字より大きい場合は, エラーになります。

cjchmodapp (アプリケーションのモード切り替え)

形式

アプリケーションをテストモードから通常モードに変更する場合

```
cjchmodapp [<サーバ名称>] [-nameserver <プロバイダURL>] -test
           -name <アプリケーション名> -mode normal
```

アプリケーションを通常モードからテストモードに変更する場合

```
cjchmodapp [<サーバ名称>] [-nameserver <プロバイダURL>]
           -name <アプリケーション名> -mode test
```

機能

J2EE アプリケーションの動作モードを, テストモードから通常モードへ, または通常モードからテストモードへ変更します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは, ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については, 「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。

-name <アプリケーション名>

モードを変更する J2EE アプリケーション名を指定します。

2. J2EE サーバで使用するコマンド

`-mode {test|normal}`

アプリケーションの動作モードを指定します。

- test

テストモードに変更します。

- normal

通常モードに変更します。このオプションを指定する場合、`-test` オプションの指定が必要です。

入力例

アプリケーションをテストモードから通常モードに変更する場合

```
cjchmodapp MyServer -test -name App1 -mode normal
```

アプリケーションを通常モードからテストモードに変更する場合

```
cjchmodapp MyServer -name App1 -mode test
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : `<アプリケーション名> -name`) やオプション名と値を非対応にすること (例 : `-nameserver <アプリケーション名> -name <プロバイダ URL>`) はできません。

コマンドの引数にサーバ名称を指定する場合、`cjsetup` コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjdeleteapp (J2EE アプリケーションの削除)

形式

アプリケーションの場合

```
cjdeleteapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
            -name <J2EEアプリケーション名>
            [-name <J2EEアプリケーション名> ...]
```

アプリケーションに含まれる EJB-JAR ファイル, WAR ファイル, または RAR ファイルの場合

```
cjdeleteapp [<サーバ名称>] [-nameserver <プロバイダURL>]
            -type {ejb|war|rar} -name <J2EEアプリケーション名>
            -resname <表示名> [-resname <表示名> ...]
```

アプリケーション中の WAR ファイルに含まれるフィルタの場合

```
cjdeleteapp [<サーバ名称>] [-nameserver <プロバイダURL>]
            -type filter -name <J2EEアプリケーション名>
            -resname <WARファイルの表示名/フィルタの表示名>
            [-resname <WARファイルの表示名/フィルタの表示名> ...]
```

機能

アプリケーション, EJB-JAR ファイル, WAR ファイル, または RAR ファイルの場合
J2EE アプリケーションまたは J2EE アプリケーションに含まれる EJB-JAR ファイル, WAR ファイル, または RAR ファイルを, 指定された J2EE サーバから削除します。また, 一度に複数のファイルの削除もできます。その際, 指定されたファイルの順に処理されます。

J2EE アプリケーション中の WAR ファイルに含まれるフィルタの場合

J2EE アプリケーション中の WAR ファイルに含まれるフィルタを削除します。削除するフィルタは, フィルタ属性ファイルに指定します。また, 一度に複数のフィルタの削除もできます。その際, 指定されたフィルタの順に処理されます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは, ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>:::<ホスト名称>:<ポート番号>

2. J2EE サーバで使用するコマンド

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-type {ejb|war|rar|filter}

J2EE アプリケーションから EJB-JAR ファイル、WAR ファイル、または RAR ファイルを削除する場合、ファイル種別を指定します。

- `ejb` : EJB-JAR ファイル
- `war` : WAR ファイル
- `rar` : RAR ファイル

J2EE アプリケーション中の WAR ファイルからフィルタを削除する場合、次のファイル種別を指定します。

- `filter` : フィルタ

-name <J2EE アプリケーション名>

削除する J2EE アプリケーション名を指定します。

-resname <表示名>

削除する EJB-JAR ファイル、WAR ファイル、RAR ファイル、またはフィルタの表示名を指定します。

入力例

アプリケーションの場合

```
cjdeleteapp MyServer -name App1
```

EJB-JAR ファイルの場合

```
cjdeleteapp MyServer -name App1 -type ejb -resname accountJsp
```

WAR ファイルの場合

```
cjdeleteapp MyServer -name App1 -type war -resname account-war
```

RAR ファイルの場合

```
cjdeleteapp MyServer -name App1 -type rar -resname account-rar
```

フィルタの場合

```
cjdeleteapp MyServer -name App1 -type filter -resname  
account-war/account-filter
```

戻り値

0 :

正常終了しました。

- 1 :
異常終了しました。
- 2 :
排他エラーによってコマンドを実行できません。
- 3 :
タイムアウトが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)

注意事項

該当する J2EE アプリケーションが開始状態の場合、このコマンドはエラー終了します。

指定された J2EE アプリケーションが存在しない場合、このコマンドはエラー終了します。

展開ディレクトリ形式のアプリケーションに対して実行した場合、J2EE サーバからアプリケーションは削除されますが、アプリケーションディレクトリは削除されません。

展開ディレクトリ形式のアプリケーションに対しては、アプリケーションに含まれる EJB-JAR ファイル、WAR ファイル、または RAR ファイルの削除はできません。

一度に複数のファイルを削除した場合、削除できないファイルがあったときでも、すべてのファイルに対して削除を試みます。一つでも削除できないファイルがあった場合、終了コードは 1 (異常終了) になります。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <表示名> -rename) やオプション名と値を非対応にすること (例 :
-nameserver <表示名> -rename <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjdeletelibjar (ライブラリ JAR の削除)

形式

```
cjdeletelibjar [<サーバ名称>] [-nameserver <プロバイダURL>]
                [-test] -name <アプリケーション名>
```

2. J2EE サーバで使用するコマンド

`-f <ライブラリJAR名> [-f <ライブラリJAR名> ...]`

機能

ライブラリ JAR を J2EE アプリケーションから削除します。複数のライブラリ JAR を指定できます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

`-nameserver <プロバイダ URL>`

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

`<プロトコル名称>::<ホスト名称>:<ポート番号>`

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

`-test`

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

`-name <アプリケーション名>`

ライブラリ JAR を削除する J2EE アプリケーション名を指定します。

`-f <ライブラリ JAR 名>`

削除対象となるライブラリ JAR 名を指定します。一度に複数のライブラリ JAR を指定できます。

入力例

```
cjdeletelibjar MyServer -name Appl -f applib.jar
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

展開ディレクトリ形式のアプリケーションに対しては使用できません。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <ライブラリ JAR 名> -f) やオプション名と値を非対応にすること (例 : -nameserver <ライブラリ JAR 名> -f <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjexportapp (J2EE アプリケーションのエクスポート)

形式

```
cjexportapp [<サーバ名称>] [-nameserver <プロバイダURL>]
-f <EARファイルのパス> -name <J2EEアプリケーション名>
[-raw|-normal]
```

機能

J2EE サーバ上の J2EE アプリケーションを EAR ファイルに出力します。実行時情報を EAR ファイルに含めることができます。

インポートした J2EE アプリケーションに cosminexus.xml が含まれている場合、cosminexus.xml を含んだ状態でエクスポートします。

インポートした J2EE アプリケーションに application.xml が含まれていない場合は、application.xml を含まない状態でエクスポートします。ただし、cjsetappprop コマンドまたは cjrenameapp コマンドの実行で application.xml が作成された場合は、application.xml を含んだ状態でエクスポートします。application.xml が作成される場合の説明については、cjsetappprop コマンドまたは cjrenameapp コマンドを参照してください。

引数

<サーバ名称>

2. J2EE サーバで使用するコマンド

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

`-nameserver <プロバイダ URL>`

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

`<プロトコル名称>::<ホスト名称>:<ポート番号>`

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

`-f <EAR ファイルのパス >`

- `-raw` を指定した場合
エクスポートする J2EE アプリケーションを含める EAR ファイルのパスを指定します。
- `-normal` を指定、または省略した場合
アプリケーションサーバのアプリケーション実行環境情報ファイル (hitachi-runtime.jar (拡張子は zip)) の出力先となるファイルのパスを指定します。

`-name <J2EE アプリケーション名 >`

エクスポートする J2EE アプリケーション名を指定します。

`-raw`

EAR ファイルに実行時情報を含めない場合に指定します。この場合、出力 EAR ファイルは、Java™ 2 Platform, Enterprise Edition に準拠したものになります。

`-normal`

EAR ファイルに実行時情報を含める場合に指定します。実行時情報を含んだ EAR ファイルは、実行時情報分だけファイルサイズが大きくなります。`-raw` オプション、`-normal` オプションのどちらも指定されていない場合、`-normal` オプションを指定した場合と同じになります。

入力例

```
cjexportapp MyServer -f App1.ear -name App1 -raw
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

- 2 :
排他エラーによってコマンドを実行できません。
- 3 :
タイムアウトが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)

注意事項

J2EE アプリケーションのエクスポート時には、実行時情報や EAR ファイルの DD に関して注意事項があります。J2EE アプリケーションのエクスポートについては、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

指定された J2EE アプリケーションが存在しない場合、このコマンドはエラー終了します。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例 : -nameserver <J2EE アプリケーション名> -name <プロバイダ URL>) はできません。

J2EE アプリケーションに含まれている cosminexus.xml の定義を変更、追加または削除した場合、cosminexus.xml は次のようにエクスポートされます。

- コメントは保持されません。
- 文字エンコードが UTF-8 になります。
- エクスポートしたアプリケーション実行環境情報ファイルや EAR ファイルに含まれるのは J2EE サーバが最後に読み込みに成功した cosminexus.xml です。

J2EE サーバが cosminexus.xml を読み込んだあとに、アプリケーションディレクトリ内の cosminexus.xml を変更しても、変更後の定義情報はエクスポートしたアプリケーション実行環境情報ファイルや EAR ファイルに含まれません。

例 : 展開ディレクトリ形式の J2EE アプリケーションをインポート後に、アプリケーションディレクトリ内の cosminexus.xml を書き換え、J2EE アプリケーションを開始しないで、その J2EE アプリケーションを EAR 形式でエクスポートした場合、EAR ファイルにはインポート後に更新したアプリケーションディレクトリ内の cosminexus.xml は含まれません。インポート時点のものと同等の

2. J2EE サーバで使用するコマンド

cosminexus.xml が含まれます。

cgjencmpsqli (CMP2.x Entity Bean 用 SQL 文の生成)

形式

```
cgjencmpsqli [サーバ名称] [-nameserver <プロバイダURL>] [-test]
              -name <アプリケーション名>
              [-resname <EJB-JARの表示名/Entity Beanの表示名>]
```

機能

CMP2.x Entity Bean 用の SQL 文を生成します。

このコマンドを実行すると次の SQL 文が生成されます。

CMP2.x Entity Bean をマッピングしたテーブルを操作する SQL 文

CMR 用のテーブルを操作する SQL 文

EJB QL から生成した SQL 文

このコマンドは、CMP2.x Entity Bean に対して次のタイミングで実行する必要があります。

Bean 間に CMR の関係がない場合

1. CMP2.x Entity Bean に対してフィールドをデータベース上のテーブルにマッピングします (cjsetapprop コマンドを使用)。
2. CMP2.x Entity Bean に対してこのコマンドを実行して SQL 文を生成します。

Bean 間に CMR の関係がある場合

1. CMR に関係するすべての CMP2.x Entity Bean に対してフィールドをデータベース上のテーブルにマッピングします (cjsetapprop コマンドを使用)。
2. CMR に関係するすべての CMP2.x Entity Bean に対してこのコマンドを実行して SQL 文を生成します。

なお、一度 SQL を生成したあとに CMR の設定を変更した場合は、デプロイをする前に、変更した CMR に関する Bean についてこのコマンドを実行して SQL 文を生成する必要があります。

アプリケーションが停止状態の場合だけ実行できます。

また、展開ディレクトリ形式の場合、コマンド実行ホストと同じホストで稼働している J2EE サーバに対して操作できます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称

がサーバ名称として使用されます。

`-nameserver <プロバイダ URL>`

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

`<プロトコル名称>::<ホスト名称>:<ポート番号>`

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

`-test`

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは, 通常モードのアプリケーションに対してコマンドが実行されます。

`-name <アプリケーション名>`

アプリケーション名を指定します。

`-resname <EJB-JAR の表示名 /Entity Bean の表示名 >`

SQL 生成対象となる Entity Bean の表示名を指定します。

省略したときは, アプリケーションに含まれるすべての CMP2.x Entity Bean が SQL 生成対象となります。SQL の生成に失敗した CMP2.x Entity Bean がある場合でも, すべての CMP2.x Entity Bean に対して SQL の生成を試みます。どれか一つでも SQL の生成が失敗した場合の終了コードは 1 となります。

入力例

```
cjgencmpsql MyServer -name App1 -resname EjbJar1/Ejb1
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドが実行できません。

3 :

タイムアウトエラーが発生しました。

9 :

管理者特権がないため, コマンドが実行できません (Windows Server 2008 x86 ,

2. J2EE サーバで使用するコマンド

Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<プロバイダ URL> -nameserver）やオプション名と値を非対応にすること（例：-name <サーバ名>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjgetappprop (アプリケーションの属性の取得)

形式

アプリケーションの場合

```
cjgetappprop [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <J2EEアプリケーション名>
              [-encoding <エンコーディング名>] -c <属性ファイルパス>
```

EJB-JAR ファイル, WAR ファイル, または RAR ファイルの場合

```
cjgetappprop [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <J2EEアプリケーション名> -type {ejb|war|rar}
              -resname <リソース表示名>[/<下位リソース表示名>]
              [-encoding <エンコーディング名>] -c <属性ファイルパス>
```

すべての情報を取得する場合

```
cjgetappprop [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <J2EEアプリケーション名> -type all
              [-encoding <エンコーディング名>] -c <属性ファイルパス>
```

機能

アプリケーション, またはそれぞれに含まれるリソース (EJB-JAR ファイルの場合: EJB-JAR ファイル, Session Bean, Entity Bean, Message-driven Bean, WAR ファイルの場合: WAR ファイル, サブレット / JSP, フィルタ) の属性, または RAR ファイルの属性を取得して, 属性ファイルを生成します。

J2EE アプリケーションとリソースの設定で使用する属性ファイルについては, マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (アプリケーション / リソース定義)」を参照してください。

展開ディレクトリ形式の場合, コマンド実行ホストと同じホストで稼働している J2EE サーバに対して操作できます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-namingserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <J2EE アプリケーション名>

取得対象の J2EE アプリケーション名を指定します。

-type {ejb|war|rar|all}

取得するリソースの種別を指定します。

- ejb : EJB-JAR ファイルまたは EJB-JAR ファイル内の EJB ファイル
- war : WAR ファイル, WAR ファイル内のサーブレット / JSP または, フィルタ
- rar : RAR ファイル
- all : すべての情報

-resname <リソース表示名>

取得対象となるリソースの表示名を指定します。

- ejb-jar : <EJB-JAR ファイルの表示名>
- ejb : <EJB-JAR ファイルの表示名>[/<Enterprise Bean の表示名>]
- war : <WAR ファイルの表示名>
- rar : <RAR ファイルの表示名>
- Servlet/JSP : <WAR ファイルの表示名>[/<サーブレット / JSP の表示名>]
- filter : <WAR の表示名>[/<フィルタの表示名>]

なお, 下位リソース表示名を指定しなかった場合は, それぞれ EJB-JAR ファイル, または WAR ファイルの指定となります。

-encoding <エンコーディング名>

属性ファイル出力時のエンコーディング名を指定します。指定できるエンコーデ

2. J2EE サーバで使用するコマンド

ング名を次に示します。

- UTF8
- UTF-8
- ASCII
- US-ASCII
- ISO-8859-1
- EUC_JP
- Shift_JIS
- SJIS
- windows-31j
- MS932
- JavaVM がサポートしているエンコーディング名

省略した場合は、JavaVM のデフォルトエンコーディング名を使用します。使用されたエンコーディング名を確認するには、このコマンドで取得した属性ファイルの先頭行の XML 宣言部分を参照してください。

```
<?xml version="<XML規格のバージョン番号>" encoding="<エンコーディング名>"?>
```

-c <属性ファイルパス>

属性ファイルの出力先パスを指定します。

入力例

アプリケーションの場合

```
cjgetappprop MyServer -name App1 -encoding Shift_JIS -c  
App1Prop.xml
```

EJB-JAR ファイルの場合

```
cjgetappprop MyServer -name App1 -type ejb -resname accountJsp/  
MyAccount -encoding UTF-8 -c MyAccountProp.xml
```

WAR ファイルの場合

```
cjgetappprop MyServer -name App1 -type war -resname account-war/  
Account -c AccountProp.xml
```

RAR ファイルの場合

```
cjgetappprop MyServer -name App1 -type rar -resname account-rar  
-c AccountProp.xml
```

すべての情報を取得する場合

```
cjgetappprop MyServer -name App1 -type all -encoding Shift_JIS  
-c App1AllProp.xml
```

戻り値

0 :

正常終了しました。

- 1 :
異常終了しました。
- 2 :
排他エラーによってコマンドを実行できません。
- 3 :
タイムアウトが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <J2EE アプリケーション名 > -name) やオプション名と値を非対応にすること (例 : -name ejb -type <J2EE アプリケーション名 >) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

指定したパスにすでに属性ファイルが存在する場合、上書きします。

cjgetstubsjar (アプリケーションの RMI-IIOP スタブおよびインタフェースの取得)

形式

```
cjgetstubsjar [<サーバ名称>] [-nameserver <プロバイダURL>]
               [-test] -name <アプリケーション名>
               -d <スタブおよびインタフェース格納先ディレクトリパス>
```

機能

J2EE アプリケーションの RMI-IIOP スタブおよびインタフェースを取得します。このコマンドは、アプリケーションの状態 (停止状態, 開始状態) に関係なく、実行できません。

引数

<サーバ名称 >

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

2. J2EE サーバで使用するコマンド

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <アプリケーション名>

RMI-IIOP スタブおよびインタフェースを取得する J2EE アプリケーション名を指定します。

-d <スタブおよびインタフェース格納先ディレクトリパス>

RMI-IIOP スタブおよびインタフェースを格納するディレクトリのパスを指定します。指定したディレクトリ下に、次のファイルが作成されます。

- stubs.jar (RMI-IIOP スタブのファイル名)
 - ?.jar (RMI-IIOP インタフェースのファイル名)
- ? には、1 からの連番が付与されます。

例えば、インタフェースが三つある場合、1.jar, 2.jar, 3.jar になります。

入力例

```
cjgetstubsjar MyServer -name Appl -d temp
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,

Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<アプリケーション名> -name）やオプション名と値を非対応にすること（例：-nameserver <アプリケーション名> -name <プロバイダ URL>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

指定したパスにすでにファイルが存在する場合、ファイルは上書きされます。

指定したパスが存在しない場合、エラーになります。

アプリケーションが停止状態で取得した RMI-IIOP スタブおよびインタフェースは、最新の状態ではない可能性があります。

-name オプションで指定したアプリケーションが一度も開始されていない場合、エラーになります。

J2EE アプリケーション内に、リモートインタフェースを持つ Enterprise Bean が存在しない場合、エラーになります。

次の J2EE アプリケーションでは、Web ブラウザから RMI-IIOP スタブおよびインタフェースを取得することはできません。このコマンドで取得してください。

- 旧バージョンでエクスポートした J2EE アプリケーションをインポートしている場合。
- 新規に作成した場合。
- cjrenameapp コマンドでアプリケーション名称を変更した場合。

cjimportapp (J2EE アプリケーションのインポート)

形式

アーカイブ形式のアプリケーションをインポートする場合

```
cjimportapp [<サーバ名称>] [-nameserver <プロバイダURL>]
            [-test] [-nodelete]
            -f <EARファイルのパス> [-f <EARファイルのパス>...]
```

展開ディレクトリ形式のアプリケーションとしてインポートする場合（アプリケーションディレクトリ）

```
cjimportapp [<サーバ名称>] [-nameserver <プロバイダURL>]
            [-test] -a <アプリケーションディレクトリパス>
```

展開ディレクトリ形式のアプリケーションとしてインポートする場合（標準のアプリ

2. J2EE サーバで使用するコマンド

ケーションファイル (拡張子 .ear) またはアプリケーションサーバのアプリケーション実行環境情報ファイル (拡張子 .zip))

```
cjimportapp [<サーバ名称>] [-nameserver <プロバイダURL>]
            [-test] [-nodelete] -f <EARファイルのパス>
            -d <展開ディレクトリパス>
```

機能

指定された J2EE サーバに J2EE アプリケーションをインポートします。なお、cosminexus.xml を含む J2EE アプリケーションをインポートした場合、cosminexus.xml もインポートされます。

実行時情報を含んだ EAR ファイルをインポートした場合、インポートした J2EE アプリケーションの実行時情報は、エクスポートしたときの J2EE アプリケーションの実行時情報と同じになります。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-nodelete

実行時情報を含む EAR ファイルをインポートする場合、開始処理などに失敗したアプリケーションを削除しないときに指定します。

このオプションを省略した場合、インポート後の開始処理などに失敗するとインポートしたアプリケーションは削除されます。

-f <EAR ファイルのパス>

インポートする J2EE アプリケーションを含む EAR ファイルのパスを指定します。一度に複数の EAR ファイルの指定もできます。

-a <アプリケーションディレクトリパス>

アプリケーションディレクトリのパスを指定します。

-d <展開ディレクトリパス>

アプリケーションを展開するサーバ側のディレクトリのフルパスを指定します。

J2EE サーバを起動したユーザに対して展開先ディレクトリの書き込み権限が必要です。

入力例

アーカイブ形式のアプリケーションをインポートする場合

```
cjimportapp MyServer -f Appl.ear
```

アプリケーションディレクトリを展開ディレクトリ形式のアプリケーションとしてインポートする場合

```
cjimportapp MyServer -a AppDirPath
```

標準のアプリケーションファイル (拡張子 .ear) またはアプリケーションサーバのアプリケーション実行環境情報ファイル (拡張子 .zip) を展開ディレクトリ形式のアプリケーションとしてインポートする場合

Windows の場合

```
cjimportapp MyServer -f Appl.zip -d
C:¥extractdir¥ApplicationDir
```

UNIX の場合

```
cjimportapp MyServer -f Appl.zip -d /extractdir/
ApplicationDir
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

8 :

実行時情報エラーが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows

Vista の場合)。

注意事項

-d オプションを使用して、EAR ファイルまたは ZIP ファイルを展開して作成されたアプリケーションディレクトリが、次の条件のどれかと一致する場合、-a オプションを使用してインポートできません。

- EJB-JAR のモジュール名が ".jar" で終わっていない。
- WAR のモジュール名が ".war" で終わっていない。
- モジュールの拡張子を除いた名称が、ほかのモジュールの拡張子を除いた名称と重複している。
- モジュールの拡張子を除いた名称が、EAR ファイル内のディレクトリ名称と重複している。

application.xml の <display-name> タグを設定していない場合、または <display-name> タグに空文字を設定している場合、J2EE アプリケーションの表示名にファイル名またはディレクトリ名が使用されます。なお、ファイル名またはディレクトリ名に汎用文字、半角英数字以外の文字が含まれている場合、その文字はアンダースコア "_" に置き換えられます。

application.xml を省略した場合、J2EE アプリケーションの表示名は次のように変換されます。

- EAR ファイルのときは、ファイル名から拡張子を除いた文字列。ただしピリオド (.) が最初にだけ現れるファイル名の場合、拡張子は除かれません。
- アプリケーションディレクトリのときは、アプリケーションディレクトリ名。

なお、変換された表示名に半角英数字 (0 ~ 9, A ~ Z, a ~ z)、アンダースコア () 以外の文字の文字がある場合、その文字はアンダースコア () に置き換えられます。

アプリケーションディレクトリを指定して展開ディレクトリ形式のアプリケーションをインポートする場合、インポートする J2EE アプリケーションに application.xml が存在するときは application.xml の <module> タグで指定したパス名から拡張子を省いた名称が EJB-JAR ディレクトリ名および WAR ディレクトリ名になります。また、インポートする J2EE アプリケーションに application.xml が存在しないときは EJB-JAR ディレクトリ名の最後は "_jar"、WAR ディレクトリ名の最後は "_war" になります。

同じアプリケーション名を持つ J2EE アプリケーションが、インポート先の J2EE サーバにすでに存在する場合、このコマンドはエラー終了します。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例: -nameserver <EAR ファイルのパス> -f <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

J2EE アプリケーションのインポート時には、このほかにも注意事項があります。J2EE アプリケーションのインポートについては、マニュアル「Cosminexus アプリケーションサーバアプリケーション設定操作ガイド」を参照してください。

インポートするアプリケーションのアプリケーション名が 255 文字より大きい場合はエラーとなります。

cjimportlibjar (ライブラリ JAR のインポート)

形式

```
cjimportlibjar [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
               -name <アプリケーション名> -f <ライブラリJARファイルパス>
               [-f <ライブラリJARファイルパス> ...]
```

機能

J2EE アプリケーションにライブラリ JAR をインポートします。また、一度に複数のファイルのインポートもできます。ライブラリ JAR のインポート先は J2EE アプリケーションのルート直下です。

ライブラリ JAR については、マニュアル「Cosminexus アプリケーションサーバシステム設計ガイド」の J2EE コンポーネントの説明を参照してください。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

2. J2EE サーバで使用するコマンド

`-name <アプリケーション名>`

ライブラリ JAR をインポートするアプリケーション名を指定します。

`-f <ライブラリ JAR ファイルパス>`

読み込み対象となるライブラリ JAR ファイルのパスを指定します。

入力例

```
cjimportlibjar MyServer -name Appl -f applib.jar
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

すでにインポートされているライブラリ JAR と同じファイル名を持つライブラリ JAR をインポートすることはできません。

展開ディレクトリ形式のアプリケーションに対しては使用できません。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <ライブラリ JAR ファイルパス> -f) やオプション名と値を非対応にすること (例: -nameserver <ライブラリ JAR ファイルパス> -f <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjlistapp (アプリケーションの一覧表示)

形式

すべてのアプリケーションの場合

```
cjlistapp [<サーバ名称>] [-nameserver <プロバイダURL>]
```

アプリケーションに含まれる EJB-JAR ファイルまたは WAR ファイルの場合

```
cjlistapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
-name <J2EEアプリケーション名> -type {ejb|war}
[-resname <表示名>]
```

アプリケーションに含まれる RAR ファイルの場合

```
cjlistapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
-name <J2EEアプリケーション名> -type rar
[-spec | -resname <表示名> -outbound| -resname <表示名>
-inbound| -resname <表示名> -listenertype <メッセージリスナのタイプ名>]
```

機能

すべてのアプリケーションの場合

すべての J2EE アプリケーションの名称、状態、およびモードの一覧を標準出力に出力します。展開ディレクトリ形式のアプリケーションの場合、アプリケーションディレクトリのパスも標準出力に出力します。

アプリケーションに含まれる EJB-JAR ファイルまたは WAR ファイルの場合

J2EE アプリケーションに含まれる EJB-JAR ファイルまたは WAR ファイルの一覧を標準出力に出力します。

`-resname` を指定すると、EJB-JAR ファイルに含まれるリソース (Session Bean, Entity Bean, Message-driven Bean) の一覧、または WAR ファイルに含まれるリソース (サーブレット / JSP, フィルタ) の一覧を標準出力に出力します。

アプリケーションに含まれる RAR ファイルの場合

J2EE アプリケーションに含まれる RAR ファイルの一覧を標準出力に出力します。

`-spec` を指定すると、RAR ファイルのコネクタアーキテクチャの仕様バージョンを標準出力に出力します。

Connector 1.5 のリソースアダプタで、`-resname` と `-outbound` を指定すると、RAR ファイルに含まれる Outbound リソースアダプタのコネクション定義識別子の一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、`-resname` と `-outbound` は指定できません。

Connector 1.5 のリソースアダプタで、`-resname` と `-inbound` を指定すると、RAR ファイルに含まれる、Inbound リソースアダプタのメッセージリスナのタイプの一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、`-resname` と `-inbound` は指定できません。

Connector 1.5 のリソースアダプタで、`-resname` と `-listenertype` を指定すると、指

2. J2EE サーバで使用するコマンド

定したメッセージリスナのアクティブ化のために設定が必要なプロパティ名の一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、`-rename` と `-listenertype` は指定できません。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

`-nameserver` <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

`-test`

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

`-name` <J2EE アプリケーション名>

J2EE アプリケーション名を指定します。

`-type` {ejb|war|rar}

表示対象となるリソースの種別を指定します。

- `ejb` : EJB-JAR ファイル
- `war` : WAR ファイル
- `rar` : RAR ファイル

`-rename` <表示名>

表示対象となるリソース (EJB-JAR ファイル、WAR ファイル、または RAR ファイル) の表示名を指定します。

`-spec`

コネクタアーキテクチャのバージョンを表示する場合に指定します。次のどちらかのコネクタアーキテクチャのバージョンが表示されます。

- Connector 1.0
- Connector 1.5

`-outbound`

RAR ファイルに含まれる Outbound リソースアダプタのコネクション定義識別子を表示する場合に、`-resname` とともに指定します。

-inbound

RAR ファイルに含まれる Inbound リソースアダプタのメッセージリスナのタイプを表示する場合に、`-resname` とともに指定します。

-listenertype <メッセージリスナのタイプ名>

指定したメッセージリスナのアクティブ化のために設定が必要なプロパティ名を表示する場合に、`-resname` とともに指定します。

入力例

すべてのアプリケーションの場合

```
cjlistapp MyServer
```

EJB-JAR ファイルの場合

```
cjlistapp MyServer -name App1 -type ejb
```

WAR ファイルの場合

```
cjlistapp MyServer -name App1 -type war -resname account-war
```

RAR ファイルの場合

```
cjlistapp MyServer -name App1 -type rar
```

出力形式

すべての J2EE アプリケーションの場合

```
<J2EEアプリケーションの状態> <J2EEアプリケーションのモード> <J2EEアプリケーション名> (path=<アプリケーションディレクトリのパス>)
<J2EEアプリケーションの状態> <J2EEアプリケーションのモード> <J2EEアプリケーション名> (path=<アプリケーションディレクトリのパス>)
...
```

```
<J2EEアプリケーションの状態>
running: 開始状態
stopped: 停止状態
stopFailure: 通常停止失敗状態
forceStopFailure: 強制停止失敗状態
blockadeFailure: 閉塞失敗状態
blockading: 閉塞中
blockade: 閉塞状態
stopping: 通常停止中
forceStopping: 強制停止中
illegal: 定義不正状態
```

```
<J2EEアプリケーションのモード>
```

```
test: テストモードで動作
(空白): 通常モードで動作
```

注 path=<アプリケーションディレクトリのパス> の情報は、展開ディレクトリ形式のアプリケーションのときだけ出力されます。アーカイブ形式のアプリケー

2. J2EE サーバで使用するコマンド

ションのときには出力されません。

EJB-JAR ファイルの場合 (-resname 指定なし)

<EJB-JARファイルの表示名>
<EJB-JARファイルの表示名>
...

EJB-JAR ファイルの場合 (-resname 指定あり)

<Enterprise Beanの表示名>
<Enterprise Beanの表示名>
...

WAR ファイルの場合 (-resname 指定なし)

<WARファイルの表示名>
<WARファイルの表示名>
...

WAR ファイルの場合 (-resname 指定あり)

<サーブレット / JSP , フィルタの表示名>
<サーブレット / JSP , フィルタの表示名>
...

RAR ファイルの場合 (-resname および -spec 指定なし)

<RARファイルの表示名>
<RARファイルの表示名>
...

RAR ファイルの場合 (-spec 指定あり)

<コネクタアーキテクチャの仕様バージョン> <RARファイルの表示名>
<コネクタアーキテクチャの仕様バージョン> <RARファイルの表示名>
...

RAR ファイルの場合 (-resname および -outbound 指定あり)

<コネクション定義識別子>
<コネクション定義識別子>
...

RAR ファイルの場合 (-resname および -inbound 指定あり)

<メッセージリスナのタイプ>
<メッセージリスナのタイプ>
...

RAR ファイルの場合 (-resname および -listenertype 指定あり)

<プロパティ名>
<プロパティ名>
...

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <J2EE アプリケーション名 > -name) やオプション名と値を非対応にすること (例 : -name ejb -type <J2EE アプリケーション名 >) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一覧は大文字小文字の区別なしで、昇順でソートされます。

cjlistlibjar (ライブラリ JAR の一覧表示)

形式

```
cjlistlibjar [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <アプリケーション名>
```

機能

J2EE アプリケーションに含まれるライブラリ JAR の一覧を標準出力に出力します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>:::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

2. J2EE サーバで使用するコマンド

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。
このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <アプリケーション名>

J2EE アプリケーション名を指定します。

入力例

```
cjlistlibjar MyServer -name App1
```

出力形式

<ライブラリJAR名>

<ライブラリJAR名>

...

戻り値

0:

正常終了しました。

1:

異常終了しました。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <アプリケーション名> -name) やオプション名と値を非対応にすること (例: -nameserver <アプリケーション名> -name <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一覧は大文字小文字の区別なしで、昇順でソートされます。

cjreloadapp (アプリケーションのリロード)

形式

```
cjreloadapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
            -name <アプリケーション名>
            [-t <強制リロード開始までのタイムアウト時間>]
```

機能

アプリケーションを停止しないで、構成するクラスおよび JSP を入れ替えます。

入れ替え後のアプリケーションに引き継がれるセッション情報の詳細については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 拡張編」を参照してください。

展開ディレクトリを使用し、開始状態およびリロードに失敗して停止状態にあるアプリケーションだけに実行できます。

このコマンドを実行する場合は、J2EE サーバ側のユーザプロパティファイル (usrconf.properties) に ejbserver.deploy.context.reload_scope プロパティを指定しておく必要があります。ユーザプロパティファイル (usrconf.properties) の詳細については、マニュアル「Cosminexus アプリケーションサーバ リファレンス 定義編 (サーバ定義)」を参照してください。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <アプリケーション名>

2. J2EE サーバで使用するコマンド

アプリケーション名を指定します。

-t < 強制リロード開始までのタイムアウト時間 >

コマンドを実行してから強制リロード開始までの待ち時間 (秒) を指定します。

0 ~ 2147483647 の整数です。

0 を指定した場合は、構成するクラスおよび JSP の入れ替え処理が終了するまでクライアントへ応答は返りません。強制リロード開始までの待ち時間 (秒) を指定しなかった場合は、強制リロードを実行しません。

入力例

```
cjreloadapp MyServer -name App1
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドが実行できません。

3 :

タイムアウトエラーが発生しました。

4 :

閉塞失敗しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : < アプリケーション名 > -name) やオプション名と値を非対応にすること (例 : -nameserver < アプリケーション名 > -name < プロバイダ URL >) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

J2EE アプリケーションの停止処理が -t オプションで指定した時間内に終了しなかった場合、サーバ管理コマンド用システムプロパティファイルの

ejbserver.rmi.request.timeout キーで指定した時間で強制終了します。

J2EE アプリケーションに cosminexus.xml が含まれている場合、cosminexus.xml の定義情報は読み込まれません。

cjrenameapp (アプリケーション名の変更)

形式

```
cjrenameapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
             -name <アプリケーション名> -newname <変更後アプリケーション名>
```

機能

指定した J2EE アプリケーションの名称を変更します。

アプリケーションの名前を変更することによって、アプリケーションの世代管理やバージョン管理を行うことができます。

なお、application.xml を省略したアプリケーションの名前を変更した場合、application.xml が作成されます。そのため、アプリケーション名を変更した J2EE アプリケーションは application.xml を含むアプリケーションになります。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <アプリケーション名>

名称を変更する J2EE アプリケーション名を指定します。

2. J2EE サーバで使用するコマンド

-newname < 変更後アプリケーション名 >

変更後の J2EE アプリケーション名を 1 文字以上で指定します。

アプリケーション名には、英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) を使用できます。

入力例

```
cjrenameapp MyServer -name App1 -newname App2
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : < アプリケーション名 > -name) やオプション名と値を非対応にすること (例 : -nameserver < アプリケーション名 > -name < プロバイダ URL >) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

このコマンドを実行する場合、名称変更の対象となるアプリケーションは停止している必要があります。

アプリケーション名称を変更した場合、lookup 名称も変更されるので lookup 名称を使用している EJB クライアントやネーミング切り替え機能などでパスを通してしている場合は、修正が必要となります。

テストモードと通常モードで、それぞれに一意のアプリケーション名称を登録できます。同一モードで同一アプリケーション名称が存在する場合は、エラーとなります。このとき、アプリケーション名の大文字・小文字は区別されません。

旧バージョンで作成されたアプリケーションについても、アプリケーション名を変更できます。

<アプリケーション名>と<変更後アプリケーション名>に同じ名前を指定した場合、正常に処理されますが、アプリケーション名は変わりません。

指定したアプリケーション名が 255 文字より大きい場合はエラーとなります。

cjreplaceapp (アプリケーションの入れ替え)

形式

```
cjreplaceapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <アプリケーション名> -f <ファイルパス>
              [-t <タイムアウト時間>] [-replaceDD]
```

機能

J2EE アプリケーションを入れ替えます。

入れ替え後の J2EE アプリケーションが `cosminexus.xml` を含む場合、入れ替え前のアプリケーションサーバ独自の定義情報を破棄してデフォルト値に戻してから、`cosminexus.xml` の定義情報で上書きします。入れ替え後の J2EE アプリケーションが `cosminexus.xml` を含まない場合、入れ替え前の J2EE アプリケーションからアプリケーションサーバ独自の定義情報を引き継ぎます。

このコマンドは、アプリケーションの状態（停止状態、開始状態）に関係なく実行できます。アプリケーションが開始されている場合、アプリケーション中のメソッドの終了を待ってアプリケーションを停止して入れ替え処理をしたあと、アプリケーションを開始します。ただし、`-t` オプションで指定された時間が経過しても停止処理が終了していない場合、アプリケーションは強制停止します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>:::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

2. J2EE サーバで使用するコマンド

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。
このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <アプリケーション名>

入れ替えるアプリケーション名を指定します。

-f <ファイルパス>

読み込み対象となるアプリケーションファイルのパスを指定します。
このオプションで指定するアプリケーションファイルは標準のアプリケーションファイル（拡張子 .ear）を指定します。

-t <タイムアウト時間>

コマンドを実行してから応答が返る時間を、0 ~ 2147483647 までの整数（単位：秒）で指定します。0 を指定した場合は、開始状態のアプリケーションの停止処理が終了するまでクライアントへ応答が返りません。
指定がない場合は 60 秒が設定されます。

-replaceDD

入れ替え前のアプリケーションの DD ファイル（application.xml, ejb-jar.xml, ra.xml, web.xml）の定義を、入れ替え後のアプリケーションに引き継がない場合に指定します。

入力例

```
cjreplaceapp MyServer -name App1 -f App1.ear
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません（Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合）。

注意事項

展開ディレクトリ形式のアプリケーションに対しては使用できません。

DD ファイル (application.xml , ejb-jar.xml , web.xml) のどれかが「なし」から「あり」に、または「あり」から「なし」になった場合、J2EE アプリケーションの入れ替えはできません。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <ファイルパス> -file) やオプション名と値を非対応にすること (例 : -nameserver <ファイルパス> -file <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

アプリケーションサーバのアプリケーション実行環境情報ファイル (拡張子 .zip) を指定した場合、エラーになります。

-name に指定されたアプリケーション名とアプリケーションファイル内の DD ファイル (application.xml) の <display-name> タグに指定されたアプリケーション名が異なる場合、エラーになります。

入れ替え前と入れ替え後の J2EE アプリケーションの構成に次のような違いがあった場合、エラーになります。

- J2EE アプリケーション内の EJB-JAR , リソースアダプタ , WAR の数が異なる場合。
- J2EE アプリケーション内の EJB-JAR ファイル , RAR ファイル , WAR ファイルの名称が異なる場合。
- -replaceDD を指定した場合に、DD ファイル (application.xml , ejb-jar.xml , ra.xml , web.xml) の定義が異なるとき。
- EJB-JAR ファイル内のホームインタフェース , コンポーネントインタフェースのメソッド定義が異なる場合。
- EJB-JAR ファイル内のホームインタフェース (ローカル , リモート) , コンポーネントインタフェース (ローカル , リモート) , ビジネスインタフェース (ローカル , リモート) のメソッド定義が異なる場合。
- 設定しているアノテーションの値を変更した場合。

J2EE アプリケーションの停止処理が -t オプションで指定した時間内に終了しなかった場合、サーバ管理コマンド用システムプロパティファイルの ejbserver.rmi.request.timeout キーで指定した時間で強制終了します。

06-70 より前のバージョンでは、入れ替え後のアプリケーションに、入れ替え前のアプリケーションのランタイム属性 だけを引き継ぎます。06-70 以降のバージョンでは、デフォルトでは、J2EE アプリケーションのすべての属性情報を引き継ぎます。06-70 より前のバージョンと同じように、ランタイム属性だけを引き継ぎたい場合は、

2. J2EE サーバで使用するコマンド

-replaceDD オプションを指定してコマンドを実行します。

注

属性ファイルには、DD ファイル (application.xml , ejb-jar.xml , ra.xml , web.xml) の定義と属性ファイル独自の定義を設定できます。属性ファイル独自の定義のことを、ランタイム属性といいます。

cjsetappprop (アプリケーションの属性設定)

形式

アプリケーションの場合

```
cjsetappprop [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <J2EEアプリケーション名> -c <属性ファイルパス>
```

EJB-JAR ファイル, WAR ファイル, または RAR ファイルの場合

```
cjsetappprop [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <J2EEアプリケーション名> -type {ejb|war|rar}
              -resname <リソース表示名>[/<下位リソース表示名>]
              -c <属性ファイルパス>
```

すべての情報を設定する場合

```
cjsetappprop [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
              -name <J2EEアプリケーション名> -type all
              -c <属性ファイルパス>
```

機能

アプリケーション, EJB-JAR ファイルに含まれるリソース (EJB-JAR ファイル, Session Bean, Entity Bean, Message-driven Bean), WAR ファイルに含まれるリソース (WAR ファイル, サブレット / JSP, フィルタ), または RAR ファイルの属性を指定したアプリケーション属性ファイルの値に変更します。

下位リソース表示名が指定されない場合は EJB-JAR ファイル, または WAR ファイルの指定になります。

J2EE アプリケーションとリソースの設定で使用する属性ファイルについては, マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (アプリケーション / リソース定義)」を参照してください。

展開ディレクトリ形式の場合, コマンド実行ホストと同じホストで稼働している J2EE サーバに対して操作できます。

cosminexus.xml を含むアプリケーションのアプリケーションサーバ独自の定義情報がこのコマンドで変更, 追加または削除された場合, アプリケーションに含まれる cosminexus.xml が更新されます。なお, 変更, 追加または削除されるアプリケーションが展開ディレクトリ形式の場合, アプリケーションディレクトリ内の cosminexus.xml が更新されます。アプリケーションサーバ独自の定義情報については, マニュアル

「Cosminexus アプリケーションサーバリファレンス 定義編（アプリケーション/リソース定義）」の `cosminexus.xml` の説明を参照してください。

`application.xml` を含まないアプリケーションに対して、次のタグの値を変更した場合、`application.xml` が作成されます。

アプリケーション属性ファイル

```
<hitachi-application-property> - <description>
<hitachi-application-property> - <icon> - <small-icon>
<hitachi-application-property> - <icon> - <large-icon>
```

WAR 属性ファイル

```
<hitachi-war-property> - <war-runtime> - <context-root>
```

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

```
<プロトコル名称>::<ホスト名称>:<ポート番号>
```

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <J2EE アプリケーション名>

設定対象の J2EE アプリケーション名を指定します。

-type {ejb|war|rar|all}

設定するリソースの種別を指定します。

- `ejb` : EJB-JAR ファイルまたは EJB-JAR ファイル内の EJB ファイルを設定
- `war` : WAR ファイルまたは WAR ファイル内のサーブレット / JSP を設定
- `rar` : RAR ファイルを設定
- `all` : すべての情報を設定

-resname <リソース表示名>

2. J2EE サーバで使用するコマンド

設定対象となるリソースの表示名を指定します。

- EJB-JAR ファイルの場合：<EJB-JAR ファイルの表示名 >
- EJB ファイルの場合：<EJB-JAR ファイルの表示名 >[/<Enterprise Bean の表示名 >]
- WAR ファイルの場合：<WAR ファイルの表示名 >
- サブレット / JSP の場合：<WAR ファイルの表示名 >[/< サブレット / JSP の表示名 >]
- フィルタの場合：<WAR の表示名 >[/< フィルタの表示名 >]
- RAR ファイルの場合：<RAR ファイルの表示名 >

-c < 属性ファイルパス >

属性ファイルの入力元パスを指定します。

入力例

アプリケーションの場合

```
cjsetappprop MyServer -name App1 -c App1Prop.xml
```

EJB-JAR ファイルの場合

```
cjsetappprop MyServer -name App1 -type ejb -resname accountJsp/MyAccount -c MyAccountProp.xml
```

WAR ファイルの場合

```
cjsetappprop MyServer -name App1 -type war -resname account-war/Account -c AccountProp.xml
```

RAR ファイルの場合

```
cjsetappprop MyServer -name App1 -type rar -resname account-rar -c AccountProp.xml
```

すべての情報を設定する場合

```
cjsetappprop MyServer -name App1 -type all -c App1AllProp.xml
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,

Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<J2EE アプリケーション名> -name）やオプション名と値を非対応にすること（例：-name ejb -type <J2EE アプリケーション名>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

指定した J2EE アプリケーション、またはリソースは存在している必要があります。

-type オプションで all を指定した場合、指定したアプリケーション属性ファイル内のリソースの表示名で変更対象とするリソースが決定します。そのため、リソースの表示名を変更することはできません。

cjstartapp (J2EE アプリケーションの開始)

形式

```
cjstartapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
           -name <J2EEアプリケーション名> [-jspc]
```

機能

J2EE アプリケーションを開始して、クライアントからのリクエストを受け取ることができるようにします。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

2. J2EE サーバで使用するコマンド

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <J2EE アプリケーション名>

開始する J2EE アプリケーション名を指定します。

-jspc

この引数を指定すると、JSP の事前コンパイルが実行されます。既存の Web アプリケーションに対して JSP の事前コンパイルを実行したあと、アプリケーションを開始する場合に指定してください。JSP の事前コンパイルでエラーが発生してもアプリケーションの開始は実行されます。

入力例

```
cjstartapp MyServer -name App1
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合) 。

注意事項

指定された J2EE アプリケーションが存在しない場合、このコマンドはエラー終了します。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例 : -nameserver <J2EE アプリケーション名> -name <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

-jspc オプションを指定する場合は、次のことに注意してください。

- -jspc オプションを指定すると、Web アプリケーションに含まれるすべての JSP ファイルがコンパイルされるため、コンパイル不要な JSP ファイルがコンパイルされてエラーになる場合があります。アプリケーション開始時の JSP 事前コンパイルでコンパイルエラーが発生した場合でも、cjstartapp コマンドは処理を続行します。
- アーカイブ形式のアプリケーションの場合、アプリケーションの開始に失敗すると、アプリケーションは停止状態になり、JSP コンパイル結果は削除されます。そのため、あらかじめアプリケーションが開始できる状態であることを確認してください。
- アプリケーション開始時の JSP 事前コンパイルを実行すると、アプリケーションに含まれる JSP の数によっては、cjstartapp コマンドの通信タイムアウトが発生する場合があります。cjstartapp コマンドで通信タイムアウトが発生した場合、J2EE サーバのメッセージログを参照してアプリケーションが正常に開始されているかを確認してください。

cjstopapp (J2EE アプリケーションの停止)

形式

通常停止を行う場合

```
cjstopapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
          -name <J2EEアプリケーション名> [-t <タイムアウト時間>]
```

通常停止がタイムアウトしたあと、強制停止を手動で行う場合

```
cjstopapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
          -name <J2EEアプリケーション名> -cancel
```

通常停止がタイムアウトしたあと、強制停止を自動で行う場合

```
cjstopapp [<サーバ名称>] [-nameserver <プロバイダURL>] [-test]
          -name <J2EEアプリケーション名> -t <タイムアウト時間> -force
```

機能

J2EE アプリケーションを停止して、クライアントからのリクエストを受け取らないようにします。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービ

2. J2EE サーバで使用するコマンド

スが稼働しているホスト名，およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。このオプションを省略したときは、通常モードのアプリケーションに対してコマンドが実行されます。

-name <J2EE アプリケーション名>

停止する J2EE アプリケーション名を指定します。

-t <タイムアウト時間>

コマンドを実行してから応答が返るタイムアウトの時間を整数（単位：秒）で指定します。

-force オプションを同時に指定する場合は 0 ~ 2147483647 を指定します。

-force オプションを指定しないで、このオプションだけを指定する場合は 0 ~ 86400 を指定します。

0 を指定した場合は、停止処理が終了するまでクライアントへ応答が返りません。

指定がない場合は 60 秒が設定されます。

-cancel

通常停止がタイムアウトしたあと、強制停止を手動で行う場合に指定します。

-force

通常停止がタイムアウトしたあと、強制停止を自動で行う場合に指定します。

入力例

```
cjstopapp MyServer -name App1
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <アプリケーション名> -name) やオプション名と値を非対応にすること (例 : -nameserver <アプリケーション名> -name <プロバイダ URL>) はできません。

通常停止を行う場合、60 秒または -t オプションで指定した時間の間に該当する J2EE アプリケーションが停止しなかったとき、このコマンドはエラー終了します。

強制停止を行う場合、J2EE アプリケーションの停止処理が -t オプションで指定した時間内に終了しなかったとき、サーバ管理コマンド用システムプロパティファイルの ejbserver.rmi.request.timeout キーで指定した時間で強制終了します。

指定された J2EE アプリケーションが存在しない場合、このコマンドはエラー終了します。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

タイムアウトエラーが発生した場合は、cjlistapp コマンドで J2EE アプリケーションの状態を確認してください。

2.4 J2EE サーバで使用するリソース操作コマンド

ここでは、J2EE サーバで使用するリソース操作コマンドについて説明します。

J2EE サーバで使用するリソース操作コマンドの一覧を、次の表に示します。

表 2-5 J2EE サーバで使用するリソース操作コマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjclearpool	コネクションプール内のコネクション削除	リソースアダプタまたはデータソースのコネクションをクリアします。	更新
cjclosecn	J2EE サーバのコネクションクローズまたはコネクションの状態表示	データソースのコネクションをクローズします。またはコネクションの状態を表示します。	更新
cjcopyres	リソースのコピー	EJB-JAR ファイル、WAR ファイル、データソースファイル、メール、または RAR ファイルのプロパティをコピーします。	更新
cjdeletejb	JavaBeans リソースの削除	インポートされている JavaBeans リソースを削除します。	更新
cjdeleteres	リソースの削除	インポート済みの EJB-JAR ファイル、WAR ファイル、データソースファイル、メール、または RAR ファイルを削除します。	更新
cjdeployrar	リソースアダプタのデプロイ	リソースアダプタをデプロイします。	更新
cjexportrar	リソースアダプタのエクスポート	J2EE サーバ上のリソースアダプタをエクスポートします。	更新
cjgetjbprop	JavaBeans リソースの属性の取得	-resname で指定された JavaBeans リソースの属性を取得して、属性ファイルを生成します。	更新
cjgetrarprop	RAR ファイルの属性の取得	デプロイ済みの RAR ファイルの属性を取得して、属性ファイルを生成します。	更新
cjgetresprop	リソースの属性の取得	EJB-JAR ファイルに含まれるリソース、WAR ファイルに含まれるリソース、データソースファイル、メール、またはデプロイ前の RAR ファイルに含まれるリソースの属性を取得して、属性ファイルを生成します。	更新
cjimportjb	JavaBeans リソースのインポート	JavaBeans リソースをインポートします。	更新

コマンド名称	分類	概要	コマンドの系統
cjimportres	リソースのインポート	EJB-JAR ファイル, WAR ファイル, データソースファイル, または RAR ファイルをインポートします。	更新
cjlistjb	JavaBeans リソースの一覧表示	すべての JavaBeans リソースの状態を標準出力に出力します。	参照
cjlistpool	コネクションプールの一覧表示	指定されたリソースアダプタ, またはすべてのリソースアダプタのコネクションプール状態を標準出力に出力します。	参照
cjlistrar	リソースアダプタの一覧表示	デPLOYされているすべてのリソースアダプタの一覧を標準出力に出力します。	参照
cjlistres	リソースの一覧表示	インポート済みの EJB-JAR ファイル, WAR ファイル, データソースファイル, メール, または RAR ファイルの一覧を標準出力に出力します。	参照
cjresumepool	メンバコネクションプールの再開始	クラスタコネクションプールのメンバコネクションプールを再開します。	更新
cjsetjbprop	JavaBeans リソースの属性設定	指定された JavaBeans リソースの属性を, 指定した JavaBeans リソース属性ファイルの値に変更します。	更新
cjsetrarprop	RAR 属性設定	デPLOY済みの RAR ファイルに含まれるリソースの属性を属性ファイルに指定された値に変更します。	更新
cjsetresprop	リソースの属性設定	EJB-JAR ファイルに含まれるリソース, WAR ファイルに含まれるリソース, データソースファイル, メール, またはデPLOY前の RAR ファイルに含まれるリソースの属性を, 属性ファイルに指定された値に変更します。	更新
cjstartjb	JavaBeans リソースの開始	JavaBeans リソースを開始します。	更新
cjstartrar	リソースアダプタの開始	リソースアダプタを開始します。	更新
cjstopjb	JavaBeans リソースの停止	開始状態の JavaBeans リソースを停止します。	更新
cjstoprar	リソースアダプタの停止	開始されているリソースアダプタを停止します。	更新
cjsuspendpool	メンバコネクションプールの一時停止	クラスタコネクションプールのメンバコネクションプールを一時停止します。	更新

2. J2EE サーバで使用するコマンド

コマンド名称	分類	概要	コマンドの系統
cjtestres	リソースの接続テスト	J2EE リソースアダプタ, メールおよびデータソースの接続テストをします。	更新
cjundeployrar	リソースアダプタのアンデプロイ	デプロイされているリソースアダプタをアンデプロイします。	更新

(凡例)

更新: 更新系コマンド。J2EE サーバの内容を更新, または構成情報を取得するコマンドです。

参照: 参照系コマンド。J2EE サーバの内容を更新しないで, 構成状態を表示するだけのコマンドです。

注

リスト表示の場合は, 参照になります。

サーバ管理コマンドの系統ごとの排他制御の条件については, マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

cjclearpool (コネクションプール内のコネクション削除)

形式

データソース (JDBC) の場合

```
cjclearpool [<サーバ名称>] [-nameserver <プロバイダURL>]
[-type jdbc] [-mode {normal | plan}]
-resname <データソース表示名>
[-resname <データソース表示名> ...]
```

リソースアダプタ (J2EE リソースアダプタとしてデプロイして使用) の場合

```
cjclearpool [<サーバ名称>] [-nameserver <プロバイダURL>]
-type connector [-mode {normal | plan}]
-resname <リソースアダプタ表示名>
[-resname <リソースアダプタ表示名> ...]
```

リソースアダプタ (J2EE アプリケーションに含めて使用) の場合

```
cjclearpool [<サーバ名称>] [-nameserver <プロバイダURL>]
-type connector [-mode {normal | plan}]
[-test]
-name <アプリケーション名>
-resname <リソースアダプタ表示名>
[-resname <リソースアダプタ表示名> ...]
```

すべてのデータソース (JDBC) およびリソースアダプタの場合

```
cjclearpool [<サーバ名称>] [-nameserver <プロバイダURL>]
[-mode {normal | plan}] -resall
```

機能

指定されたデータソース (JDBC) またはリソースアダプタのコネクションをクリアしま

す。一度に複数のデータソース (JDBC) またはリソースアダプタのコネクションを削除できます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-type {jdbc|connector}

コネクションプールの種別を指定します。省略した場合は jdbc が設定されます。

- jdbc : JDBC 用のプール
- connector : リソースアダプタ用のプール

-mode {normal|plan}

使用中のコネクションに対するクリア動作種別を指定します。未使用のコネクションはすべてクリア対象になります。省略した場合は normal が設定されます。

- normal (通常モード)
ユーザが使用中のコネクションはプールの外へ移して、プール内からはクリアされます。ただし、データベースとの接続はそのまま再利用はできません。
- plan (計画モード)
ユーザが使用中のコネクションはクリアしません。ただし、再利用できません。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。省略した場合は通常モードのアプリケーションに対してコマンドを実行します。

-name <アプリケーション名>

リソースアダプタがデプロイされているアプリケーション名を指定します。コネクションプール種別が connector の場合に指定できます。省略した場合は、J2EE リソースアダプタとしてデプロイして使用したリソースアダプタのコネクションを削除します。

-resname <データソース表示名> または <リソースアダプタ表示名>

クリア対象となるデータソース (JDBC) またはリソースアダプタの表示名を指定し

2. J2EE サーバで使用するコマンド

ます。クラスタコネクションプールのルートリソースアダプタの表示名は指定できません。指定した場合、異常終了します。クラスタコネクションプールのメンバリソースアダプタの表示名を指定した場合には、コネクションプールの状態が開始状態である必要があります。開始状態でない場合、異常終了します。

リソースアダプタ表示名の指定方法は、リソースアダプタのバージョンによって次のように異なります。

- Connector 1.0 の場合
 < リソースアダプタの表示名 >
- Connector 1.5 の場合
 < リソースアダプタの表示名 >!< コネクション定義識別子 >

-resall

すべてのデータソース (JDBC) およびリソースアダプタを削除対象にする場合に指定します。

なお、-type、-name、または -resname と同時に指定できません。

入力例

データソース (JDBC) の場合 (通常モード)

```
cjclearpool MyServer -resname JdbcConnection
```

リソースアダプタ (J2EE リソースアダプタとしてデプロイして使用) の場合

```
cjclearpool MyServer -type connector -mode plan -resname  
RarConnection
```

リソースアダプタ (J2EE アプリケーションに含めて使用) の場合

```
cjclearpool MyServer -type connector -mode plan -name Appl  
-resname RarConnection
```

すべてのデータソース (JDBC) およびリソースアダプタの場合 (計画モード)

```
cjclearpool MyServer -mode plan -resall
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,

Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<データソース表示名> -resname）やオプション名と値を非対応にすること（例：-nameserver <データソース表示名> -resname <プロバイダ URL>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一度に複数のデータソース（JDBC）またはリソースアダプタを削除した場合、削除できないデータソース（JDBC）またはリソースアダプタがあったときでも、すべてのデータソース（JDBC）またはリソースアダプタに対して削除を試みます。

通常モードの場合、コネクションプールから取り除いた使用中コネクションは、コネクション数としてカウントされません。そのため、コネクションプール内のコネクションとコネクションプールから取り除いた使用中コネクションの合計がコネクションプールの最大値を超える場合があります。

cjclosecn (J2EE サーバのコネクションクローズまたはコネクションの状態表示)

形式

JDBC コネクション一覧を表示する場合

指定されたデータソースを対象とする場合

```
cjclosecn [<サーバ名称>] [-nameserver <プロバイダURL>]
-l [-verbose] -resname <データソース表示名>
[-resname <データソース表示名> ...]
```

プールに登録されたすべてのデータソースを対象とする場合

```
cjclosecn [<サーバ名称>] [-nameserver <プロバイダURL>]
-l [-verbose] -resall
```

タイムアウト時間を指定してコネクションをクローズする場合

指定されたデータソースを対象とする場合

```
cjclosecn [<サーバ名称>] [-nameserver <プロバイダURL>]
-t <seconds> -resname <データソース表示名>
[-resname <データソース表示名> ...]
```

プールに登録されたすべてのデータソースを対象とする場合

```
cjclosecn [<サーバ名称>] [-nameserver <プロバイダURL>]
```

2. J2EE サーバで使用するコマンド

```
-t <seconds> -resall
```

コネクション ID を指定してコネクションをクローズする場合

```
cjclosecn [<サーバ名称>] [-nameserver <プロバイダURL>]
          -resname <データソースの表示名>
          -cid <コネクションID> [-cid <コネクションID> ...]
```

機能

指定されたデータソースまたはプールに登録されたすべてのデータソースを対象として、使用中のコネクションの中で、取得時からの経過時間が指定時間を超過しているものを強制的にクローズします。このコマンドでは、次の機能を使用できます。

JDBC コネクション一覧を表示する

指定されたデータソースまたはプールに登録されたすべてのデータソースを対象として、JDBC コネクション一覧を表示します。

-verbose を指定しない場合

使用中の JDBC コネクション一覧を表示します。データソースを指定する場合、一度に複数のデータソースを指定できます。

-verbose を指定した場合

使用中および未使用のすべての JDBC コネクション一覧を表示します。

タイムアウト時間を指定してコネクションをクローズする

指定されたデータソースまたはプールに登録されたすべてのデータソースを対象として、使用中のコネクションの中で、取得時からの経過時間が指定時間を超過しているものを強制的にクローズします。データソースを指定する場合、一度に複数のデータソースを指定できます。

コネクション ID を指定してコネクションをクローズする

指定されたデータソースを対象として、JDBC コネクションの中で、コネクション ID を指定されたコネクションを強制的にクローズします。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を、次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください

さい。

-l

使用中のコネクション情報の一覧を表示します。

-t <seconds>

コネクションを解放する際のタイムアウト時間を指定します。指定時間を超過したコネクションが解放対象となります。指定は秒単位で、指定できる値は 0 指定値 2147483647 の範囲になります。

-verbose

使用中のコネクションおよび使用中でないコネクションを表示対象とします。

-resname <データソース表示名>

クローズ対象となるデータソース名称を指定します。一度に複数のデータソースを指定できます。

-resall

すべてのデータソースの使用中コネクションが対象となります。
-resname と同時に指定できません。

-cid <コネクション ID>

-l オプションで表示した一覧情報の中から、-resname で指定したリソースが取得したコネクションを ID 番号で指定します。コネクション ID は、16 進数で指定します。ID の指定は 8 けたまで指定できます。

入力例

JDBC コネクション一覧を表示する場合（指定されたデータソースを対象とする場合）

```
cjclosecn MyServer -l -resname JdbcDbpsv
```

タイムアウト時間を指定してコネクションをクローズする場合（指定されたデータソースを対象とする場合）

```
cjclosecn MyServer -t 200 -resname JdbcDbpsv -resname JdbcDbpsv2
```

コネクション ID を指定してコネクションをクローズする場合

```
cjclosecn MyServer -resname JdbcDbpsv -cid 657ed
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

2. J2EE サーバで使用するコマンド

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合)。

注意事項

このコマンドの機能は、ベーシックモードで実行する場合だけ有効となります。1.4 モードで実行すると、サーバエラーとなります。

cjclearpool コマンドでプールから削除したコネクションは、このコマンドの対象にはなりません。

コマンド実行時、未決着の JDBC トランザクションはタイムアウトになります。また、未決着の JDBC トランザクションはロールバックします。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) であれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: `-resname <データソース表示名>`) やオプション名と値を非対応にすること (例: `-cid <データソース表示名> -resname <コネクション ID>`) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一度に複数のデータソースを指定する場合、存在しないデータソースがあったときでも、すべてのデータソースに対して、JDBC コネクション一覧の表示またはコネクションのクローズを試みます。

コネクション ID を指定する場合、未使用のコネクションを指定したときはエラーとなります。

cjcopyres (リソースのコピー)

形式

```
cjcopyres [<サーバ名称>] [-nameserver <プロバイダURL>]
          -type {ejb|war|datasource|mail|rar}
          -src <コピー元表示名> -dst <コピー先表示名>
```

機能

リソース (EJB-JAR ファイル, WAR ファイル, データソースファイル, メール, または RAR ファイル) のプロパティをコピーします。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-type {ejb|war|datasource|mail|rar}

コピー対象となるリソースの種別を指定します。

- ejb : EJB-JAR ファイル
- war : WAR ファイル
- datasource : データソースファイル
- mail : メール
- rar : RAR ファイル

-src <コピー元表示名>

コピー元となる EJB-JAR ファイル, WAR ファイル, データソースファイル, メール, または RAR ファイルの表示名を指定します。

-dst <コピー先表示名>

コピー先となる EJB-JAR ファイル, WAR ファイル, データソースファイル, メール, または RAR ファイルの表示名を指定します。

表示名には, 英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) を使用できます。

入力例

EJB-JAR ファイルの場合

```
cjcopyres MyServer -type ejb -src accountJsp -dst accountJsp2
```

WAR ファイルの場合

```
cjcopyres MyServer -type war -src account-war -dst account_war2
```

データソースファイルの場合

```
cjcopyres MyServer -type datasource -src JdbcDbpsv -dst JdbcDbpsv2
```

2. J2EE サーバで使用するコマンド

メールの場合

```
cjcopyres MyServer -type mail -src Mail -dst Mail2
```

RAR ファイルの場合

```
cjcopyres MyServer -type rar -src account-ra -dst account-ra2
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)。

注意事項

コピー先に指定する表示名が存在してはいけません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称
と大文字・小文字を一致させた文字列を指定する必要があります。

コピー先表示名に指定したデータソースファイル、メール、または RAR ファイルの
表示名が 240 文字より大きい場合はエラーとなります。

cjdeletejb (JavaBeans リソースの削除)

形式

```
cjdeletejb [<サーバ名称>] [-nameserver <プロバイダURL>]  
-resname <JavaBeansリソース表示名>  
[-resname <JavaBeansリソース表示名> ...]
```

機能

インポートされている JavaBeans リソースを削除します。

一度に複数の JavaBeans リソースを削除できます。削除できない JavaBeans リソース
があった場合でも、すべての JavaBeans リソースに対して削除しようと試みます。

一度でも開始された JavaBeans リソースは、J2EE サーバを再起動するまで削除できません。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

省略時は NamingService が返すデフォルト値を使用します。

-rename <JavaBeans リソース表示名>

削除対象となる JavaBeans リソースの表示名を指定します。

入力例

```
cjdeletejb MyServer -rename MyJavaBeans
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドが実行できません。

3:

タイムアウトエラーが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<プロバイダ URL> -nameserver）やオプション名と値を非対応にすること（例：-resname <プロバイダ URL> -nameserver <JavaBeans リソース表示名>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjdeleteres（リソースの削除）

形式

```
cjdeleteres [<サーバ名称>] [-nameserver <プロバイダURL>]
             -type {ejb|war|datasource|mail|rar} -resname <表示名>
             [-resname <表示名> ...]
```

機能

インポート済みの EJB-JAR ファイル、WAR ファイル、データソースファイル、メール、または RAR ファイルを削除します。一度に複数のファイルの削除もできます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>.:<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-type {ejb|war|datasource|mail|rar}

削除するリソースの種別を指定します。

- ejb : EJB-JAR ファイル
- war : WAR ファイル
- datasource : データソースファイル

- mail : メール
- rar : RAR ファイル

-resname <表示名>

削除対象となる EJB-JAR ファイル, WAR ファイル, データソースファイル, メール, または RAR ファイルの表示名を指定します。

入力例

EJB-JAR ファイルの場合

```
cjdeleteres MyServer -type ejb -resname accountJsp
```

WAR ファイルの場合

```
cjdeleteres MyServer -type war -resname account-war
```

データソースファイルの場合

```
cjdeleteres MyServer -type datasource -resname JdbcDbpsv
```

メールの場合

```
cjdeleteres MyServer -type mail -resname Mail
```

RAR ファイルの場合

```
cjdeleteres MyServer -type rar -resname account-ra
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

一度に複数のファイルを削除した場合, 削除できないファイルがあったときでも, すべてに対して削除を試みます。

いったん開始または停止した単体 RAR ファイルは, J2EE サーバを再起動しないと削

2. J2EE サーバで使用するコマンド

除できません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjdeployrar (リソースアダプタのデプロイ)

形式

```
cjdeployrar [<サーバ名称>] [-nameserver <プロバイダURL>]
             -resname <リソースアダプタ名> [-resname <リソースアダプタ名>
             ...]
```

機能

指定されたリソースアダプタをデプロイします。まだデプロイされていないリソースアダプタが対象となります。すでにアプリケーションに追加されているリソースアダプタの表示名と同じ表示名を持つリソースアダプタを指定することはできません。一度に複数のリソースアダプタをデプロイできます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-resname <リソースアダプタ名>

リソースアダプタ名を指定します。

入力例

```
cjdeployrar MyServer -resname account-ra
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

8:

デプロイに成功しましたが、開始に失敗しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) であれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <リソースアダプタ名> -resname) やオプション名と値を非対応にすること (例: -resname <プロバイダ URL> -nameserver <リソースアダプタ名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一度に複数のリソースアダプタをデプロイする場合、追加できないリソースアダプタがあるときでも、すべてのリソースアダプタの追加を試みます。

開始済みの実行時情報を含んだリソースアダプタを指定すると、デプロイ完了後に自動開始処理が行われます。自動開始処理に失敗した場合、J2EE リソースアダプタは削除されません。

cjexportrar (リソースアダプタのエクスポート)

形式

```
cjexportrar [<サーバ名称>] [-nameserver <プロバイダURL>]
           -f <ファイルパス> -resname <リソースアダプタ名> [-raw|-normal]
```

機能

J2EE サーバ上のリソースアダプタを RAR ファイルに出力します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-f<ファイルパス>

RAR ファイルの出力先となるファイルのパスを指定します。

-resname <リソースアダプタ名>

エクスポートするリソースアダプタ名を指定します。

-raw

J2EE リソースアダプタファイルを生成します。

-normal

日立固有のリソースアダプタ拡張 DD ファイル（ファイル名：hitachi-ra.xml）を追加し、J2EE リソースアダプタファイルを生成します。-raw オプション、-normal オプションのどちらも指定されていない場合は、-normal オプションを指定した場合と同じになります。

入力例

```
cjexportrar MyServer -f res1.rar -resname Res1
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) であれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <リソースアダプタ名> -resname) やオプション名と値を非対応にすること (例 : -resname <プロバイダ URL> -nameserver <リソースアダプタ名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjgetjbprop (JavaBeans リソースの属性の取得)

形式

```
cjgetjbprop [<サーバ名称>] [-nameserver <プロバイダURL>]
-resname <JavaBeansリソース表示名>
[-encoding <エンコーディング名>] -c <属性ファイルパス>
```

機能

-resname で指定された JavaBeans リソースの属性を取得し、属性ファイルを生成します。

JavaBeans リソース属性ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (アプリケーション/リソース定義)」を参照してください。

指定したパスにすでにファイルが存在する場合、上書きします。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

2. J2EE サーバで使用するコマンド

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-rename <JavaBeans リソース表示名 >

取得対象となる JavaBeans リソースの表示名を指定します。

-encoding <エンコーディング名 >

属性ファイル出力時のエンコーディング名を指定します。指定できるエンコーディング名を次に示します。

- UTF8
- UTF-8
- ASCII
- US-ASCII
- ISO-8859-1
- EUC_JP
- Shift_JIS
- SJIS
- windows-31j
- MS932
- JavaVM がサポートしているエンコーディング名

省略した場合は、JavaVM のデフォルトエンコーディング名を使用します。使用されたエンコーディング名を確認するには、cjgetjbprop コマンドで取得した属性ファイルの先頭行の XML 宣言部分を参照してください。

```
<?xml version="<XML規格のバージョン番号>" encoding="<エンコーディング名>"?>
```

-c <属性ファイルパス >

属性ファイルの出力先パスを指定します。

入力例

```
cjgetjbprop MyServer -rename MyJavaBeansName -encoding Shift_JIS  
-c MyJavaBeansProp.xml
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドが実行できません。

3 :

タイムアウトエラーが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例 : -resname <プロバイダ URL> -nameserver <JavaBeans リソース表示名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjgetrarprop (RAR ファイルの属性の取得)

形式

```
cjgetrarprop [<サーバ名称>] [-nameserver <プロバイダURL>]
             -resname <リソースアダプタ表示名>
             [-encoding <エンコーディング名>] -c <属性ファイルパス>
```

機能

-resname に指定された J2EE リソースとしてデプロイ済みの RAR ファイルの属性を取得して、属性ファイルを生成します。

RAR ファイルの属性を設定するのは、Connector 属性ファイルです。Connector 属性ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス定義編 (アプリケーション / リソース定義)」を参照してください。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

2. J2EE サーバで使用するコマンド

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-resname <リソースアダプタ表示名>

取得対象となるリソース（デプロイ済みのリソースアダプタ）の表示名を指定します。

-encoding <エンコーディング名>

属性ファイル出力時のエンコーディング名を指定します。指定できるエンコーディング名を次に示します。

- UTF8
- UTF-8
- ASCII
- US-ASCII
- ISO-8859-1
- EUC_JP
- Shift_JIS
- SJIS
- windows-31j
- MS932
- JavaVM がサポートしているエンコーディング名

省略した場合は、JavaVM のデフォルトエンコーディング名を使用します。使用されたエンコーディング名を確認するには、このコマンドで取得した属性ファイルの先頭行の XML 宣言部分を参照してください。

```
<?xml version="<XML規格のバージョン番号>" encoding="<エンコーディング名>"?>
```

-c <属性ファイルパス>

属性ファイルの出力先パスを指定します。

入力例

```
cjgetrarprop MyServer -resname account-rar -encoding Shift_JIS -c MyRarprop.xml
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

- 2 :
排他エラーによってコマンドを実行できません。
- 3 :
タイムアウトが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <リソースアダプタ表示名> -resname) やオプション名と値を非対応にすること (例 : -resname <属性ファイルパス> -c <リソースアダプタ表示名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

指定したパスにすでに属性ファイルが存在する場合、上書きします。

cjgetresprop (リソースの属性の取得)

形式

EJB-JAR ファイルおよび WAR ファイルの場合

```
cjgetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type {ejb|war}
               -resname <リソース表示名> [/<下位リソース表示名>]
               [-encoding エンコーディング名] -c <属性ファイルパス>
```

データソースファイルの場合

```
cjgetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type datasource -resname <データソース表示名>
               [-encoding エンコーディング名] -c <属性ファイルパス>
```

メールの場合

```
cjgetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type mail -resname <メール表示名>
               [-encoding エンコーディング名] -c <属性ファイルパス>
```

RAR ファイルの場合

```
cjgetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type rar -resname <リソースアダプタ表示名>
               [-encoding エンコーディング名] -c <属性ファイルパス>
```

2. J2EE サーバで使用するコマンド

機能

EJB-JAR ファイルに含まれるリソース (EJB-JAR ファイル, Session Bean, Entity Bean, Message-driven Bean), WAR ファイルに含まれるリソース (WAR ファイル, サブレット / JSP, フィルタ), データソースファイル, メール, またはデプロイ前の RAR ファイルに含まれるリソースの属性を取得して, 属性ファイルを生成します。

下位リソース表示名が指定されない場合は, EJB-JAR ファイルまたは WAR ファイルの指定となります。

J2EE アプリケーションとリソースの設定で使用する属性ファイルについては, マニュアル「Cosminexus アプリケーションサーバ リファレンス 定義編 (アプリケーション / リソース定義)」を参照してください。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは, ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については, 「2.1(2) プロバイダ URL について」を参照してください。

-type {ejb|war|datasource|mail|rar}

生成する属性ファイルの種別を指定します。

- ejb : EJB-JAR ファイルまたは EJB-JAR ファイル中のリソースの属性ファイル
- war : WAR ファイルまたは WAR ファイル中のリソースの属性ファイル
- datasource : データソース属性ファイル
- mail : メール属性ファイル
- rar : デプロイ前の RAR ファイルの属性ファイル

-rename <リソース表示名, データソース表示名, メール表示名またはリソースアダプタ表示名>

取得対象となるリソースの表示名を指定します。

- EJB-JAR ファイルの場合 : <EJB-JAR ファイルの表示名 >
- EJB ファイルの場合 : <EJB-JAR ファイルの表示名 >[/<EJB ファイルの表示名 >]
- WAR ファイルの場合 : <WAR ファイルの表示名 >

- サブレット / JSP の場合 : <WAR ファイルの表示名 >[/< サブレット / JSP の表示名 >]
- フィルタの場合 : <WAR の表示名 >[/< フィルタの表示名 >]
- RAR ファイルの場合 : <RAR ファイルの表示名 >

-encoding < エンコーディング名 >

属性ファイル出力時のエンコーディング名を指定します。指定できるエンコーディング名を次に示します。

- UTF8
- UTF-8
- ASCII
- US-ASCII
- ISO-8859-1
- EUC_JP
- Shift_JIS
- SJIS
- windows-31j
- MS932
- JavaVM がサポートしているエンコーディング名

省略した場合は、JavaVM のデフォルトエンコーディング名を使用します。使用されたエンコーディング名を確認するには、このコマンドで取得した属性ファイルの先頭行の XML 宣言部分を参照してください。

```
<?xml version="<XML規格のバージョン番号>" encoding="<エンコーディング名>"?>
```

-c < 属性ファイルパス >

属性ファイルの出力先パスを指定します。

入力例

EJB-JAR ファイルの場合

```
cjgetresprop MyServer -type ejb -resname accountJsp/MyAccount
-encoding Shift_JIS -c MyAccountProp.xml
```

WAR ファイルの場合

```
cjgetresprop MyServer -type war -resname account-war/Account
-encoding UTF-8 -c AccountProp.xml
```

データソースファイルの場合

```
cjgetresprop MyServer -type datasource -resname JdbcDbpsv -c
JdbcDbpsvProp.xml
```

メールの場合

```
cjgetresprop MyServer -type mail -resname Mail -c MailProp.xml
```

2. J2EE サーバで使用するコマンド

RAR ファイルの場合

```
cjgetresprop MyServer -type rar -resname account-rar -encoding  
Shift_JIS -c AccountProp.xml
```

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。
- 2 :
排他エラーによってコマンドを実行できません。
- 3 :
タイムアウトが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <J2EE アプリケーション名> -name) やオプション名と値を非対応にすること (例: -name ejb -type <J2EE アプリケーション名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

指定したパスにすでに属性ファイルが存在する場合、上書きします。

cjimportjb (JavaBeans リソースのインポート)

形式

JavaBeans リソースのインポートをする場合

```
cjimportjb [<サーバ名称>] [-nameserver <プロバイダURL>]  
-f <JARファイルパス> -c <属性ファイルパス>
```

引数で指定したディレクトリにある JavaBeans リソースのクラスのインポートをする場合

```
cjimportjb [<サーバ名称>] [-nameserver <プロバイダURL>]
```

```
-d <JavaBeansリソースのクラスが存在するディレクトリパス>
-c <属性ファイルパス>
```

機能

JavaBeans リソースのインポートをする場合

JavaBeans リソースをインポートします。

すでにインポートした JavaBeans リソースと同じ表示名、または同じ実装クラス名を持つ JavaBeans リソースをインポートすることはできません。

JavaBeans リソースをインポートし、属性ファイルに指定された値を設定します。

JavaBeans リソース属性ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（アプリケーション/リソース定義）」を参照してください。

引数で指定したディレクトリにある JavaBeans リソースのクラスのインポートをする場合

引数で指定したディレクトリにある JavaBeans リソースのクラスをインポートします。

すでにインポートした JavaBeans リソースと同じ表示名、または同じ実装クラス名を持つ JavaBeans リソースをインポートすることはできません。

JavaBeans リソースをインポートし、属性ファイルに指定された値を設定します。

JavaBeans リソース属性ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（アプリケーション/リソース定義）」を参照してください。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

```
<プロトコル名称>::<ホスト名称>:<ポート番号>
```

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-f <JAR ファイルパス>

読み込み対象となるファイル（JavaBeans リソースを含む JAR ファイル）のパスを指定します。

2. J2EE サーバで使用するコマンド

`-d` <JavaBeans リソースのクラスが存在するディレクトリパス >

JavaBeans リソースのクラスが存在するディレクトリのパスを指定します。

指定されたディレクトリ下に存在するものすべてを JavaBeans リソースとしてインポートします。

`-c` <属性ファイルパス >

属性ファイルの入力元パスを指定します。

入力例

形式 1

```
cjimportjb MyServer -f Myjavabeans.jar -c Myjavabeansprop.xml
```

形式 2

```
cjimportjb MyServer -d MydirectoryPath -c Myjavabeansprop.xml
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドが実行できません。

3:

タイムアウトエラーが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <アプリケーション名> -name) やオプション名と値を非対応にすること (例: -f <属性ファイルパス> -c <JAR ファイルパス>) はできません。

コマンドの引数にサーバ名称を指定する場合、`cjsetup` コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

JavaBeans リソースの表示名が 240 文字より大きい場合はエラーとなります。

cjimportres (リソースのインポート)

形式

EJB-JAR ファイル, WAR ファイル, または RAR ファイルの場合

```
cjimportres [<サーバ名称>] [-nameserver <プロバイダURL>]
            -type {ejb|war|rar} -f <ファイルパス>
            [-f <ファイルパス> ...]
```

データソースの場合

```
cjimportres [<サーバ名称>] [-nameserver <プロバイダURL>]
            -type datasource -resname <データソース表示名>
            -c <データソース設定ファイルパス>
            -f <JDBCドライバファイルパス>
            [-f <JDBCドライバファイルパス> ...]
```

機能

リソース (EJB-JAR ファイル, WAR ファイル, データソースファイル, または RAR ファイル) をインポートします。また, 一度に複数のファイルのインポートもできます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは, ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については, 「2.1(2) プロバイダ URL について」を参照してください。

-type {ejb|war|datasource|rar}

インポートするファイルのファイル種別を指定します。

- ejb : EJB-JAR ファイル
- war : WAR ファイル
- datasource : JDBC ドライバファイルおよびデータソース設定ファイル
- rar : RAR ファイル

-resname <データソース表示名>

インポートするデータソースに付与する名前を指定します。名前には英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) を使用できます。

2. J2EE サーバで使用するコマンド

`-c` <データソース設定ファイルパス>

読み込み対象となるデータソース設定ファイルのパスを指定します。

`-f` <ファイルパスまたは JDBC ドライバファイルパス>

読み込み対象となるファイル (EJB-JAR ファイル, WAR ファイル, JDBC ドライバ, または RAR ファイル) のパスを指定します。

JDBC ドライバが複数のファイルで構成されている場合, `-f` オプションを複数回指定してください。

入力例

EJB-JAR ファイルの場合

```
cjimportres MyServer -type ejb -f account.jar
```

WAR ファイルの場合

```
cjimportres MyServer -type war -f account.war
```

データソースファイルの場合

```
cjimportres MyServer -type datasource -resname JdbcDbpsv -c  
JdbcDbpsvConf.xml -f JdbcDbpsv.jar
```

RAR ファイルの場合

```
cjimportres MyServer -type rar -f account.rar
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため, コマンドが実行できません (Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合, コマンド名の直後にする必要があります。そのほかの引数は, サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし, オプション名と値の順序の入れ替え

(例: <データソース表示名> -resname) やオプション名と値を非対応にすること
(例: -nameserver <データソース表示名> -resname <プロバイダ URL>) はできません。

DD ファイルの <display-name> タグを設定していない場合、または <display-name> タグに空文字を設定している場合、リソースの表示名にファイル名が使用されます。なお、ファイル名に汎用文字、半角英数字以外の文字が含まれている場合、その文字はアンダースコア "_" に置き換えられます。

Deployment Descriptor を省略した場合、リソースの表示名は次のように変換されます。

- EJB-JAR ファイル (EJB3.0 以降) のときは、ファイル名から拡張子 (.jar) を除いた文字列。ただし
- WAR ファイル (Servlet2.5 以降) のときは、ファイル名から拡張子を除いた文字列。

なお、ピリオド (.) が最初にだけ現れるファイル名の場合、拡張子は除かれませんが、変換された表示名に半角英数字 (0 ~ 9, A ~ Z, a ~ z), アンダースコア () 以外の文字の文字がある場合、その文字はアンダースコア () に置き換えられます。

RAR ファイルの場合、DD ファイルは省略できません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

すでにインポートされている EJB-JAR ファイルと同じ表示名を持つ EJB-JAR ファイルはインポートできません。

すでにインポートされている WAR ファイルと同じ表示名を持つ WAR ファイルはインポートできません。

すでにインポートされている RAR ファイルと同じ表示名を持つ RAR ファイルはインポートできません。

すでにインポートされているデータソース、生成済みのメールと同じ表示名でデータソースをインポートできません。

ネイティブライブラリを含む RAR ファイルはインポートできません。

EJB-JAR に表示名が指定されていない場合、EJB-JAR ファイルのファイル名 (.jar を除く) を EJB-JAR ファイルの表示名として使用します。

RAR ファイルの場合も同様になります。

WAR ファイルの場合も同様にファイル名を表示名としますが、英数字、アンダースコア以外の文字はアンダースコアに置き換えます。置換対象文字が連続する場合は一つのアンダースコアにまとめられます。

一度に複数のファイルをインポートした場合、インポートできないファイルがあったときでも、すべてのファイルに対してインポートを試みます。

Enterprise Bean (EJB-JAR) のインポート時には、EJB-JAR ファイルの DD に関し

2. J2EE サーバで使用するコマンド

で注意事項があります。Enterprise Bean (EJB-JAR) のインポートについては、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

サーブレットと JSP のインポート時には、WAR ファイルの DD に関して注意事項があります。サーブレットと JSP のインポートについては、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

リソースアダプタのインポート時には、RAR ファイルの DD に関して注意事項があります。リソースアダプタのインポートについては、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

インポートする RAR ファイルの表示名またはデータソースの表示名が 240 文字より大きい場合はエラーとなります。

cjlistjb (JavaBeans リソースの一覧表示)

形式

```
cjlistjb [<サーバ名称>] [-nameserver <プロバイダURL>]
```

機能

すべての JavaBeans リソースの状態を標準出力に出力します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

出力形式

```
<JavaBeansリソースの状態><JavaBeansリソース表示名>  
<JavaBeansリソースの状態><JavaBeansリソース表示名>  
...
```

JavaBeans リソースの状態には次のどちらかが入ります。

- running
開始状態
- stopped
停止状態

入力例

```
cjlistjb MyServer
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

3 :

タイムアウトエラーが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) に指定します。ただし、オプション名と値の順序の入れ替え (例 : <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例 : -nameserver <サーバ名称>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjlistpool (コネクションプールの一覧表示)

形式

J2EE リソースアダプタとしてデプロイされているリソースアダプタを指定する場合

```
cjlistpool [<サーバ名称>] [-nameserver <プロバイダURL>]
           -resname <リソースアダプタの表示名>
           [-resname <リソースアダプタの表示名> ...]
```

アプリケーションに含めて使用されるリソースアダプタを指定する場合

```
cjlistpool [<サーバ名称>] [-nameserver <プロバイダURL>]
           [-test]
```

2. J2EE サーバで使用するコマンド

```
-name <アプリケーション名>  
-resname <リソースアダプタの表示名>  
[-resname <リソースアダプタの表示名> ...]
```

すべてのリソースアダプタを指定する場合

```
cjlistpool [<サーバ名称>] [-nameserver <プロバイダURL>] -resall
```

機能

J2EE リソースアダプタとしてデプロイされているリソースアダプタを指定する場合

指定された J2EE リソースアダプタのコネクションプールの情報を標準出力に出力します。

一度に複数のリソースアダプタのコネクションプールの情報を表示できます。表示できないリソースアダプタがあった場合でも、すべてのリソースアダプタに対して表示を試みます。

アプリケーションに含めて開始されているリソースアダプタを指定する場合

指定されたアプリケーションに含まれるリソースアダプタのコネクションプールの状態を標準出力に出力します。

一度に複数のリソースアダプタのコネクションプールの状態を表示できます。表示できないリソースアダプタがあった場合でも、すべてのリソースアダプタに対して表示を試みます。

すべてのリソースアダプタを指定する場合

起動されているすべてのリソースアダプタのコネクションプールの情報を標準出力に出力します。

なお、無効なコネクションプールは出力されません。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-resname <リソースアダプタの表示名>

表示対象とするリソースアダプタの表示名を指定します。

Connector 1.5 の場合、< リソースアダプタの表示名 >!< コネクション定義識別子 > で指定します。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。省略した場合、通常モードのアプリケーションに対してコマンドを実行します。

-name < アプリケーション名 >

表示対象とするリソースアダプタを含んでいるアプリケーション名を指定します。省略した場合、J2EE リソースアダプタとしてデプロイされているリソースアダプタが対象となります。

-resall

すべてのリソースアダプタを表示対象とする場合に指定します。

出力形式

Resource Name = [<アプリケーション名>:]<リソースアダプタの表示名>	} コネクションプール情報
Connection Definition = <コネクション定義識別子>* ¹	
Minimum Size = <コネクション数の最小値>	
Maximum Size = <コネクション数の最大値>	
Current Size = <コネクション数の現在値>	} メンバコネクションプール情報* ⁴
Root Resource Adapter Name = <ルートリソースアダプタ名>* ² , Status = <コネクションプールの状態>* ³	

Active Connection = <使用中のコネクション数>	} 使用中コネクション
Hashcode = <コネクションハッシュコード> rootAP = <ルートAP情報>	
Create Time = <コネクション生成時刻>	
Lapsed Time (s) = <コネクション生成経過時間 (単位: 秒)>	
Get Time = <コネクション取得時刻>	
Lapsed Time (s) = <コネクション使用経過時間 (単位: 秒)>	
Connection ID = <コネクションID>* ¹	} 未使用コネクション
:	

Free Connection = <未使用のコネクション数>	} 未使用コネクション
Hashcode = <コネクションハッシュコード>	
Create Time = <コネクション生成時刻>	
Lapsed Time (s) = <コネクション生成経過時間 (単位: 秒)>	
Last Use Time = <コネクション返却時刻>* ¹ ,	
Lapsed Time (s) = <コネクション未使用時間 (単位: 秒)>	
Connection ID = <コネクションID>* ¹	} 未使用コネクション
:	
:	

注※1 コネクションID、コネクション返却時刻およびコネクション識別子がない場合には、「N/A」が出力されます。

注※2 ルートアダプタが存在しない場合や開始していない場合には、「N/A」が出力されます。

注※3 cjlistrarコマンドと同じ状態が出力されます。コネクションプールの状態が取得できない、または状態が不正な場合には、「invalid」が出力されます。

注※4 メンバコネクションプール情報は、メンバリソースアダプタの場合だけ出力されます。

2. J2EE サーバで使用するコマンド

<コネクションプールの状態>

表示されるコネクションプールの状態には、次のどれかの値が入ります。

- running：開始状態
- runningReserved：開始予約状態
- resumingAutomatically：自動再開中状態
- resumingManually：手動再開中状態
- blockedAutomatically：自動閉塞状態
- blockedManually：手動閉塞状態
- suspendedAutomatically：自動一時停止状態
- suspendedAutomaticallyReserved：自動一時停止予約状態
- suspendedManually：手動一時停止状態
- suspendedManuallyReserved：手動一時停止予約状態
- invalid：不正状態
- N/A：メンバリソースアダプタでない場合

<コネクション ID>

コネクション ID を生成できない場合は「N/A」を出力します。

コネクション ID に関する詳細については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 基本・開発編（コンテナ共通機能）」を参照してください。

入力例

形式 1

```
cjlistpool MyServer -resname Rar1
```

形式 2

```
cjlistpool MyServer -name App1 -resname Rar1
```

形式 3

```
cjlistpool MyServer -resall
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

3:

タイムアウトエラーが発生しました。

9:

管理者特権がないため、コマンドが実行できません（Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows

Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<プロバイダ URL> -nameserver）やオプション名と値を非対応にすること（例：-nameserver <リソースアダプタの表示名> -resname <プロバイダ URL>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

コネクションプールの状態に invalid が出力された場合、メンバリソースアダプタをアンデプロイして再度デプロイしたあとに、再開始する必要があります。対処できない場合は、保守情報を取得して保守員に連絡してください。

cjlistrar (リソースアダプタの一覧表示)

形式

リソースアダプタの状態表示の場合

```
cjlistrar [<サーバ名称>] [-nameserver <プロバイダURL>]
          [-clusterpool] [-spec]
```

コネクション定義識別子の一覧表示の場合

```
cjlistrar [<サーバ名称>] [-nameserver <プロバイダURL>]
          -resname <リソースアダプタの表示名> -outbound
```

メッセージリスナのタイプの一覧表示の場合

```
cjlistrar [<サーバ名称>] [-nameserver <プロバイダURL>]
          -resname <リソースアダプタの表示名> -inbound
```

アクティブ化のために設定が必要なプロパティ名の一覧表示の場合

```
cjlistrar [<サーバ名称>] [-nameserver <プロバイダURL>]
          -resname <リソースアダプタの表示名>
          -listenertype <メッセージリスナのタイプ名>
```

機能

リソースアダプタの状態表示の場合

デプロイされているすべてのリソースアダプタについて、リソースアダプタ名とリソースアダプタの状態を標準出力に出力します。また、リソースアダプタがクラスタコネクションプールのメンバリソースアダプタの場合は、コネクションプールの状態も表示します。

-spec を指定すると、RAR ファイルのコネクタアーキテクチャの仕様バージョンを標準出力に出力します。

2. J2EE サーバで使用するコマンド

コネクション定義識別子の一覧表示の場合

Connector 1.5 のリソースアダプタで、`-resname` と `-outbound` を指定すると、デプロイされている RAR ファイルに含まれる Outbound リソースアダプタのコネクション定義識別子の一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、`-resname` と `-outbound` は指定できません。

Connector 1.5 のリソースアダプタで、`-resname` と `-inbound` を指定すると、デプロイされている RAR ファイルに含まれる、Inbound リソースアダプタのメッセージリスナのタイプの一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、`-resname` と `-inbound` は指定できません。

Connector 1.5 のリソースアダプタで、`-resname` と `-listenertype` を指定すると、指定したメッセージリスナのアクティブ化のために設定が必要なプロパティ名の一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、`-resname` と `-listenertype` は指定できません。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

`-nameserver` <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

`-clusterpool`

コネクションプールの状態を表示する場合に指定します。コネクションプールのクラスタ化機能を使用している場合に有効になります。

`-spec`

コネクタアーキテクチャのバージョンを表示する場合に指定します。

次のどちらかのコネクタアーキテクチャのバージョンが表示されます。

- Connector 1.0
- Connector 1.5

`-resname` <リソースアダプタの表示名>

表示対象となるリソースアダプタ名を指定します。

`-outbound`

RAR ファイルに含まれる Outbound リソースアダプタのコネクション定義識別子を表示する場合に、`-resname` とともに指定します。

`-inbound`

RAR ファイルに含まれる Inbound リソースアダプタのメッセージリスナのタイプを表示する場合に、`-resname` とともに指定します。

`-listenertype` <メッセージリスナのタイプ名>

指定したメッセージリスナのアクティブ化のために設定が必要なプロパティ名を表示する場合に、`-resname` とともに指定します。

出力形式

`-clusterpool` を指定しない場合

```
<リソースアダプタの状態> <リソースアダプタ名>
<リソースアダプタの状態> <リソースアダプタ名>
. . .
```

<リソースアダプタの状態> には、次のどちらかの値が入ります。

- `running` : 開始状態
- `stopped` : 停止状態

`-clusterpool` を指定した場合

```
<リソースアダプタの状態> <コネクションプールの状態> <リソースアダプタ名>
<リソースアダプタの状態> <コネクションプールの状態> <リソースアダプタ名>
. . .
```

<コネクションプールの状態> には、次のどれかの値が入ります。

- `running` : 開始状態
- `runningReserved` : 開始予約状態
- `resumingAutomatically` : 自動再開中状態
- `resumingManually` : 手動再開中状態
- `blockedAutomatically` : 自動閉塞状態
- `blockedManually` : 手動閉塞状態
- `suspendedAutomatically` : 自動一時停止状態
- `suspendedAutomaticallyReserved` : 自動一時停止予約状態
- `suspendedManually` : 手動一時停止状態
- `suspendedManuallyReserved` : 手動一時停止予約状態
- `invalid` : 不正状態
- `N/A` : メンバリソースアダプタでない場合

`-spec` を指定した場合

```
<リソースアダプタの状態> <コネクタアーキテクチャの仕様バージョン> <リソースアダプタ名>
<リソースアダプタの状態> <コネクタアーキテクチャの仕様バージョン> <リソースアダプタ名>
```

2. J2EE サーバで使用するコマンド

スアダプタ名>

...

-clusterpool と -spec を指定した場合

<リソースアダプタの状態> <コネクションプールの状態> <コネクタアーキテク
チャの仕様バージョン> <リソースアダプタ名>

<リソースアダプタの状態> <コネクションプールの状態> <コネクタアーキテク
チャの仕様バージョン> <リソースアダプタ名>

...

-rename と -outbound を指定した場合

<コネクション定義識別子>

<コネクション定義識別子>

...

-rename と -inbound を指定した場合

<メッセージリスナのタイプ>

<メッセージリスナのタイプ>

...

-rename と -listenertype を指定した場合

<プロパティ名>

<プロパティ名>

...

入力例

```
cjlistrar MyServer
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) に指定します。ただし、オプション名と値の順序の入れ替え (例: <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例:

-nameserver <サーバ名称>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一覧は大文字小文字の区別なしで、昇順でソートされます。

コネクションプールの状態に invalid が出力された場合、メンバリソースアダプタをアンデプロイして再度デプロイしたあとに、再開始する必要があります。対処できない場合は、保守情報を取得して保守員に連絡してください。

cjlistres (リソースの一覧表示)

形式

RAR ファイル以外のリソースの場合

```
cjlistres [<サーバ名称>] [-nameserver <プロバイダURL>]
          -type {ejb|war|datasource|mail} [-resname <表示名>]
```

RAR ファイルのリソースの場合

```
cjlistres [<サーバ名称>] [-nameserver <プロバイダURL>] -type rar
[-spec]
          -resname <表示名> -outbound |
          -resname <表示名> -inbound |
          -resname <表示名> -listenertype <メッセージリスナのタイプ名>]
```

機能

RAR ファイル以外のリソースの場合

インポート済みの EJB-JAR ファイル、WAR ファイル、データソースファイル、またはメールの一覧を標準出力に出力します。

RAR ファイルのリソースの場合

インポート済みの RAR ファイルの一覧を標準出力に出力します。

-spec を指定すると、RAR ファイルのコネクタアーキテクチャの仕様バージョンを標準出力に出力します。

Connector 1.5 のリソースアダプタで、-resname と -outbound を指定すると、RAR ファイルに含まれる Outbound リソースアダプタのコネクション定義識別子の一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、-resname と -outbound は指定できません。

Connector 1.5 のリソースアダプタで、-resname と -inbound を指定すると、RAR ファイルに含まれる、Inbound リソースアダプタのメッセージリスナのタイプの一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、-resname と -inbound は指定できません。

Connector 1.5 のリソースアダプタで、-resname と -listenertype を指定すると、指定したメッセージリスナのアクティブ化のために設定が必要なプロパティ名の一覧を標準出力に出力します。Connector 1.0 のリソースアダプタでは、-resname と

2. J2EE サーバで使用するコマンド

-listenertype は指定できません。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>.:<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-type {ejb|war|datasource|mail|rar}

表示対象となるリソースの種別を指定します。

- ejb : EJB-JAR ファイル
- war : WAR ファイル
- datasource : データソースファイル
- mail : メール
- rar : RAR ファイル

-rename <表示名>

表示対象となるリソース (EJB-JAR ファイル、WAR ファイル、または RAR ファイル) の表示名を指定します。

EJB-JAR ファイル、WAR ファイル、または RAR ファイルの場合に -rename を指定すると、それぞれに含まれるリソース (EJB-JAR ファイル: Enterprise Bean、WAR ファイル: サブレット / JSP または、フィルタ、RAR ファイル: RAR ファイルのコネクション定義識別子) の一覧を標準出力に出力します。

-spec

コネクタアーキテクチャのバージョンを表示する場合に指定します。

次のどちらかのコネクタアーキテクチャのバージョンが表示されます。

- Connector 1.0
- Connector 1.5

-outbound

RAR ファイルに含まれる Outbound リソースアダプタのコネクション定義識別子を表示する場合に、-rename とともに指定します。

-inbound

RAR ファイルに含まれる Inbound リソースアダプタのメッセージリスナのタイプを表示する場合に、`-resname` とともに指定します。

`-listenertype` <メッセージリスナのタイプ名>

指定したメッセージリスナのアクティブ化のために設定が必要なプロパティ名を表示する場合に、`-resname` とともに指定します。

入力例

EJB-JAR ファイルの場合

```
cjlistres MyServer -type ejb
```

WAR ファイルの場合

```
cjlistres MyServer -type war
```

データソースファイルの場合

```
cjlistres MyServer -type datasource
```

メールの場合

```
cjlistres MyServer -type mail
```

RAR ファイルの場合

```
cjlistres MyServer -type rar
```

出力形式

EJB-JAR ファイルの場合 (`-resname` 指定なし)

```
<EJB-JARファイルの表示名>
<EJB-JARファイルの表示名>
...
```

EJB-JAR ファイルの場合 (`-resname` 指定あり)

```
<Enterprise Beanの表示名>
<Enterprise Beanの表示名>
...
```

WAR ファイルの場合 (`-resname` 指定なし)

```
<WARファイルの表示名>
<WARファイルの表示名>
...
```

WAR ファイルの場合 (`-resname` 指定あり)

```
<サーブレット/JSP, フィルタの表示名>
<サーブレット/JSP, フィルタの表示名>
...
```

データソースファイルの場合

```
<データソースファイルの表示名>
<データソースファイルの表示名>
...
```

メールの場合

2. J2EE サーバで使用するコマンド

<メールの表示名>
<メールの表示名>
...

RAR ファイルの場合 (-resname および -spec 指定なし)

<RARファイルの表示名>
<RARファイルの表示名>
...

RAR ファイルの場合 (-spec 指定あり)

<コネクタアーキテクチャの仕様バージョン> <RARファイルの表示名>
<コネクタアーキテクチャの仕様バージョン> <RARファイルの表示名>
...

RAR ファイルの場合 (-resname および -outbound 指定あり)

<コネクション定義識別子>
<コネクション定義識別子>
...

RAR ファイルの場合 (-resname および -inbound 指定あり)

<メッセージリスナのタイプ>
<メッセージリスナのタイプ>
...

RAR ファイルの場合 (-resname および -listenertype 指定あり)

<プロパティ名>
<プロパティ名>
...

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows
Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること

(例: `-nameserver ejb -type <プロバイダ URL>`) はできません。

コマンドの引数にサーバ名称を指定する場合、`cjsetup` コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一覧は大文字小文字の区別なしで、昇順でソートされます。

cjresumepool (メンバコネクションプールの再開始)

形式

```
cjresumepool [<サーバ名称>] [-nameserver <プロバイダURL>]
             -resname <リソースアダプタ表示名>
             [-resname <リソースアダプタ表示名> ...]
```

機能

指定したクラスタコネクションプールのメンバコネクションプールを再開します。一度に複数のメンバコネクションプールの再開もできます。再開に失敗したメンバコネクションプールがあった場合でも、すべてのメンバコネクションプールに対して再開を試みます。一つでも再開できなかった場合は、異常終了します。

このコマンドは 1.4 モードで実行します。ベーシックモードで実行した場合は、異常終了となります。

リソースアダプタが開始状態の場合、このコマンドを実行すると、メンバコネクションプールは手動再開中状態になります。次に、メンバコネクションプールのウォーミングアップ機能が有効な場合は、メンバコネクションプール内にコネクションをプールし、その後、メンバコネクションプールは開始状態となります。

リソースアダプタが停止状態の場合、このコマンドを実行すると、メンバコネクションプールは開始予約状態になります。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>:::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

2. J2EE サーバで使用するコマンド

-resname < リソースアダプタ表示名 >

再開対象となるメンバリソースアダプタの表示名を指定します。

クラスタコネクションプールのメンバリソースアダプタでないリソースアダプタを指定した場合、異常終了となります。

この引数は、指定したリソースアダプタの開始 / 停止状態に関係なく、このコマンド自体を実行できます。ただし、メンバコネクションプールの状態によって、コマンドを実行できる場合とできない場合があります。詳細は次の表を参照してください。

メンバコネクションプールの状態	コマンドの実行可否
自動一時停止状態 手動一時停止状態 自動一時停止予約状態 手動一時停止予約状態	実行可能
開始状態 開始予約状態 自動再開中状態 手動再開中状態 自動閉塞状態 手動閉塞状態	実行不可

入力例

```
cjresumepool MyServer -resname  
DB_Connector_for_Oracle_ClusterPool_Member
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドが実行できません。

3 :

タイムアウトエラーが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合) 。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え

(例: <アプリケーション名> -name) やオプション名と値を非対応にすること (例: -nameserver <アプリケーション名> -name <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合, cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjsetjbprop (JavaBeans リソースの属性設定)

形式

```
cjsetjbprop [<サーバ名称>] [-nameserver <プロバイダURL>]
             -resname <JavaBeansリソース表示名> -c <属性ファイルパス>
```

機能

-resname で指定された JavaBeans リソースの属性を, 指定した JavaBeans リソース属性ファイルの値に変更します。JavaBeans リソース属性ファイルの詳細については, マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (アプリケーション/リソース定義)」を参照してください。

指定した JavaBeans リソースは存在している必要があります。

JavaBeans リソースの表示名は, 属性ファイルの <display-name> タグで指定したものに置き換わります。

すでにスタート状態にある JavaBeans リソースの属性は変更できません。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは, ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>:::<ホスト名称>:<ポート番号>

指定内容の詳細については, 「2.1(2) プロバイダ URL について」を参照してください。

-resname <JavaBeans リソース表示名>

設定対象の JavaBeans リソース表示名を指定します。

-c <属性ファイルパス>

2. J2EE サーバで使用するコマンド

属性ファイルの入力元パスを指定します。

入力例

```
cjsetjbprop MyServer -resname MyJavaBEansName -c  
MyJavaBeansPorp.xml
```

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。
- 2 :
排他エラーによってコマンドが実行できません。
- 3 :
タイムアウトエラーが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <アプリケーション名> -name) やオプション名と値を非対応にすること (例 : -resname <属性ファイルパス> -c <JavaBeans リソース表示名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjsetrarprop (RAR 属性設定)

形式

```
cjsetrarprop [ <サーバ名称> ] [ -nameserver <プロバイダURL> ]  
-resname <リソースアダプタ表示名> -c <属性ファイルパス>
```

機能

-resname に指定されたデプロイ済みの RAR ファイルに含まれるリソースの属性を属性ファイルに指定された値に変更します。

RAR ファイルの属性を設定するのは、Connector 属性ファイルです。Connector 属性

ファイルについては、マニュアル「Cosminexus アプリケーションサーバ リファレンス 定義編（アプリケーション/リソース定義）」を参照してください。

リソースの表示名は、属性ファイルの <display-name> タグで指定したものに置き換わります。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-rename <リソースアダプタ表示名>

設定対象リソース（デプロイ済みのリソースアダプタ）の表示名を指定します。

-c <属性ファイルパス>

属性ファイルの入力元パスを指定します。

入力例

```
cjsetrarprop MyServer -rename account-rar -c MyRarprop.xml
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません（Windows Server 2008 x86 ,

2. J2EE サーバで使用するコマンド

Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<リソースアダプタ表示名> -resname）やオプション名と値を非対応にすること（例：-resname <属性ファイルパス> -c <リソースアダプタ表示名>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjsetresprop (リソースの属性設定)

形式

EJB-JAR ファイルおよび WAR ファイルの場合

```
cjsetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type {ejb|war}
               -resname <リソース表示名> [ / <下位リソース表示名> ]
               -c <属性ファイルパス>
```

データソースファイルの場合

```
cjsetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type datasource -resname <データソース表示名>
               -c <属性ファイルパス>
```

メールの場合

```
cjsetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type mail -resname <メール表示名> -c <属性ファイルパス>
```

RAR ファイルの場合

```
cjsetresprop [<サーバ名称>] [-nameserver <プロバイダURL>]
               -type rar -resname <リソースアダプタ表示名>
               -c <属性ファイルパス>
```

機能

EJB-JAR ファイルに含まれるリソース（EJB-JAR ファイル，Session Bean，Entity Bean，Message-driven Bean），WAR ファイルに含まれるリソース（WAR ファイル，サーブレット / JSP，フィルタ），データソースファイル，メール，またはデプロイ前の RAR ファイルに含まれるリソースの属性を属性ファイルに指定された値に変更します。

下位リソース表示名が指定されない場合は EJB-JAR ファイル，WAR ファイルの指定となります。

J2EE アプリケーションとリソースの設定で使用する属性ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（アプリケーション/リソース定義）」を参照してください。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-type {ejb|war|datasource|mail|rar}

設定するリソースの種別を指定します。

- ejb : EJB-JAR ファイルまたは EJB-JAR ファイル中のリソースの属性
- war : WAR ファイルまたは WAR ファイル中のリソースの属性
- datasource : データソースファイル
- mail : メール
- rar : デプロイ前の RAR ファイルの属性

-resname <リソース表示名, データソース表示名, メール表示名またはリソースアダプタ表示名>

設定対象リソースの表示名を指定します。

- EJB-JAR ファイルの場合 : <EJB-JAR ファイルの表示名 >
- EJB ファイルの場合 : <EJB-JAR ファイルの表示名 >[/<Enterprise Bean の表示名 >]
- WAR ファイルの場合 : <WAR ファイルの表示名 >
- サブレット / JSP の場合 : <WAR ファイルの表示名 >[/<サブレット / JSP の表示名 >]
- フィルタの場合 : <WAR の表示名 >[/<フィルタの表示名 >]
- RAR ファイルの場合 : <RAR ファイルの表示名 >

-c <属性ファイルパス>

属性ファイルの入力元パスを指定します。

2. J2EE サーバで使用するコマンド

入力例

EJB-JAR ファイルの場合

```
cjsetresprop MyServer -type ejb -resname accountJsp/MyAccount -c  
MyAccountProp.xml
```

WAR ファイルの場合

```
cjsetresprop MyServer -type war -resname account-war/Account -c  
AccountProp.xml
```

データソースファイルの場合

```
cjsetresprop MyServer -type datasource -resname JdbcDbpsv -c  
JdbcDbpsvProp.xml
```

メールの場合

```
cjsetresprop MyServer -type mail -resname Mail -c MailProp.xml
```

RAR ファイルの場合

```
cjsetresprop MyServer -type rar -resname account-ra -c  
AccountProp.xml
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <リソース表示名> -resname) やオプション名と値を非対応にすること (例 : -resname ejb -type <リソース表示名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

属性ファイルに設定された security-role/link-to の値は無視されます。指定されたりソースが Session Bean, Entity Bean, または WAR ファイルの場合は runtime の値も無視されます。

データソース表示名は、属性ファイルの <display-name> タグで指定したものに置き換わります。

リソース表示名は、属性ファイルの <display-name> タグで指定したものに置き換わります。

データソースの場合、指定した表示名は存在している必要があります。

メールの場合、指定した表示名が存在しないときは新規登録として扱います。新規登録では、英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) を使用できます。また、既存のものに設定する場合、メール表示名は、プロパティファイルの <display-name> タグで指定したものに置き換わります。

cjstartjb (JavaBeans リソースの開始)

形式

```
cjstartjb [<サーバ名称>] [-nameserver <プロバイダURL>]
          -resname <JavaBeansリソース表示名>
```

機能

JavaBeans リソースを開始します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-resname <JavaBeans リソース表示名>

JavaBeans リソースの表示名を指定します。

2. J2EE サーバで使用するコマンド

入力例

```
cjstartjb MyServer -resname javabeansname
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドが実行できません。

3:

タイムアウトエラーが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例: -nameserver JavaBeans <リソース表示名> -resname<プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjstartrar (リソースアダプタの開始)

形式

```
cjstartrar [<サーバ名称>] [-nameserver <プロバイダURL>]  
-resname <リソースアダプタ表示名>
```

機能

指定されたデプロイ済みのリソースアダプタを開始します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

`-nameserver <プロバイダ URL>`

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

`<プロトコル名称>::<ホスト名称>:<ポート番号>`

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

`-rename <リソースアダプタ表示名>`

デプロイされたリソースアダプタの表示名を指定します。

入力例

```
cjstartrar MyServer -rename account-ra
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: `<リソースアダプタ表示名> -rename`) やオプション名と値を非対応にすること (例: `-rename <プロバイダ URL> -nameserver <リソースアダプタ表示名>`) はできません。

2. J2EE サーバで使用するコマンド

コマンドの引数にサーバ名称を指定する場合、`cjsetup` コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjstopjb (JavaBeans リソースの停止)

形式

```
cjstopjb [<サーバ名称>] [-nameserver <プロバイダURL>]
        -resname <JavaBeansリソース表示名>
```

機能

開始状態の JavaBeans リソースを停止します。

アプリケーションから使用される JavaBeans リソースを停止する場合は、先に JavaBeans リソースを使用するアプリケーションを停止しておく必要があります。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-resname <JavaBeans リソース表示名>

JavaBeans リソースの表示名を指定します。

入力例

```
cjstopjb MyServer -resname javabeansname
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドが実行できません。

3:

タイムアウトエラーが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <アプリケーション名> -name) やオプション名と値を非対応にすること (例 : -nameserver <JavaBeans リソース表示名> -resname <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjstoprar (リソースアダプタの停止)

形式

```
cjstoprar [<サーバ名称>] [-nameserver <プロバイダURL>]
          -resname <リソースアダプタ表示名>
```

機能

開始されているリソースアダプタを停止します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください

2. J2EE サーバで使用するコマンド

さい。

`-rename <リソースアダプタ表示名>`
停止するリソースアダプタ名を指定します。

入力例

```
cjstoprar MyServer -rename account-ra
```

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。
- 2 :
排他エラーによってコマンドを実行できません。
- 3 :
タイムアウトが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : `<リソースアダプタ表示名> -rename`) やオプション名と値を非対応にすること (例 : `-rename <プロバイダ URL> -nameserver <リソースアダプタ表示名>`) はできません。

コマンドの引数にサーバ名称を指定する場合、`cjsetup` コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjsuspendpool (メンバコネクションプールの一時停止)

形式

```
cjsuspendpool [<サーバ名称>] [-nameserver <プロバイダURL>]  
                  -rename <リソースアダプタ表示名>  
                  [-rename <リソースアダプタ表示名> ...]
```

機能

クラスタコネクションプールのメンバコネクションプールを一時停止します。一度に複数のメンバコネクションプールの一時停止もできます。一時停止に失敗したメンバコネクションプールがある場合でも、すべてのメンバコネクションプールに対して一時停止を試みます。どれか一つでも一時停止できなかった場合は異常終了となります。

1.4 モードで実行する場合だけ有効です。ベーシックモードで実行した場合には、異常終了となります。

リソースアダプタが開始状態のときに、このコマンドを実行すると、メンバコネクションプールは手動閉塞状態になり、コマンドは終了します。その後、J2EE サーバでは一時停止処理が行われます。一時停止処理が完了すると、メンバコネクションプールは手動一時停止状態になります。

リソースアダプタが停止状態のときに、このコマンドを実行すると、メンバコネクションプールは手動一時停止予約状態になります。

引数

< サーバ名称 >

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver < プロバイダ URL >

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>:::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-rename < リソースアダプタ表示名 >

一時停止対象となるメンバリソースアダプタの表示名を指定します。

クラスタコネクションプールのメンバリソースアダプタの表示名以外を指定した場合、異常終了となります。

この引数は、指定したリソースアダプタの開始 / 停止状態に関係なく、このコマンド自体を実行できます。ただし、メンバコネクションプールの状態によって、コマンドを実行できる場合とできない場合があります。詳細は次の表を参照してください。

メンバコネクションプールの状態	コマンドの実行可否
開始状態 開始予約状態 自動一時停止状態 自動一時停止予約状態	実行可能

2. J2EE サーバで使用するコマンド

メンバコネクションプールの状態	コマンドの実行可否
手動一時停止状態 手動一時停止予約状態 自動再開中状態 手動再開中状態 自動閉塞状態 手動閉塞状態	実行不可

入力例

```
cjsuspendpool MyServer -resname  
DB_Connector_for_Oracle_ClusterPool_Member
```

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。
- 2 :
排他エラーによってコマンドが実行できません。
- 3 :
タイムアウトエラーが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合) 。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <データソース表示名> -resname) やオプション名と値を非対応にすること (例: -nameserver <データソース表示名> -resname <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjtestres (リソースの接続テスト)

形式

データソースおよびメールの場合

```

cjtestres [<サーバ名称>] [-nameserver <プロバイダURL>]
          -type {datasource|mail}
          -resname <表示名> [-resname <表示名> ...]

```

J2EE リソースアダプタとしてデプロイして使用するリソースアダプタの場合

```

cjtestres [<サーバ名称>] [-nameserver <プロバイダURL>]
          -type rar
          -resname <表示名> [-resname <表示名> ...]

```

アプリケーションに含めて使用するリソースアダプタの場合

```

cjtestres [<サーバ名称>] [-nameserver <プロバイダURL>]
          [-test] -name <アプリケーション名> -type rar
          -resname <表示名> [-resname <表示名> ...]

```

機能

J2EE リソースアダプタ、メールおよびデータソースの接続テストをします。

リソースアダプタにコネクション定義が複数ある場合、すべてのコネクション定義に対して接続テストを行います。コネクション定義のどれかでエラーが発生した場合でもすべてのコネクション定義に対して接続テストを実行します。

なお、Inbound リソースアダプタの接続テストはできません。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-test

テストモードのアプリケーションに対してコマンドを実行する場合に指定します。省略した場合、通常モードのアプリケーションに対してコマンドを実行します。

2. J2EE サーバで使用するコマンド

-name <アプリケーション名>

アプリケーションに含まれるリソースアダプタの接続テストを行う場合に、アプリケーション名を指定します。

-type {datasource|mail|rar}

接続テストの対象となるリソースの種別を指定します。

- datasource : データソースファイル
- mail : メール
- rar : RAR ファイル

-resname <表示名>

接続テストの対象となるデータソースファイル、メール、または RAR ファイルの表示名を指定します。このオプションは、複数回指定できます。

入力例

データソースの場合

```
cjtestres -type datasource -resname Myds1
```

メールの場合

```
cjtestres -type mail -resname Mymail1
```

J2EE リソースアダプタとしてデプロイして使用するリソースアダプタの場合

```
cjtestres -type rar -resname Myrar1
```

アプリケーションに含めて使用するリソースアダプタの場合

```
cjtestres -name App1 -type rar -resname Myrar1
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<表示名> -rename）やオプション名と値を非対応にすること（例：-nameserver <表示名> -rename <プロバイダ URL>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjundeployrar (リソースアダプタのアンデプロイ)

形式

```
cjundeployrar [<サーバ名称>] [-nameserver <プロバイダURL>]
               -rename <リソースアダプタ表示名>
               [-rename <リソースアダプタ名> ...]
```

機能

指定されたリソースアダプタをアンデプロイします。デプロイされているリソースアダプタが対象となります。一度に複数のリソースアダプタをアンデプロイできます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-rename <リソースアダプタ表示名>

リソースアダプタ名を指定します。

入力例

```
cjundeployrar MyServer -rename account-ra
```

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。
- 2 :
排他エラーによってコマンドを実行できません。
- 3 :
タイムアウトが発生しました。
- 9 :
管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
Vista の場合) 。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要がありません。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <リソースアダプタ名> -resname) やオプション名と値を非対応にすること (例 : -resname <プロバイダ URL> -nameserver <リソースアダプタ名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一度に複数のリソースアダプタをアンデプロイする場合、削除できないリソースアダプタがあるときでも、すべてのリソースアダプタの削除を試みます。

一度でも開始・接続テストを試みた J2EE リソースアダプタは、J2EE サーバを再起動するまでアンデプロイできません。

2.5 J2EE サーバで使用するそのほかのコマンド

ここでは、J2EE サーバで使用するそのほかのコマンドについて説明します。

J2EE サーバで使用するそのほかのコマンドの一覧を、次の表に示します。

表 2-6 J2EE サーバで使用するそのほかのコマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjaddsec	ユーザとロールの追加	ユーザまたはロールを追加します。	更新
cjdeletesec	ユーザとロールの削除	ユーザまたはロールを削除します。	更新
cjlistsec	ユーザとロールの一覧表示	ユーザまたはロールの一覧を標準出力に出力します。	参照
cjlistthread	スレッド情報の表示	処理中のスレッドに関する情報を出力します。 テストモードで実行できます。	参照
cjlisttrn	稼働中の J2EE サーバのトランザクション情報の表示	J2EE サーバで稼働しているトランザクション情報の一覧を標準出力に出力します。	特権
cjlisttrnfile	停止中の J2EE サーバのトランザクション情報の表示	停止中の J2EE サーバのトランザクション情報の一覧を標準出力に出力します。	-
cjmapsec	ユーザとロールのマッピング	ロールにユーザを追加します。	更新
cjresetsv	サーバ管理コマンドの排他強制解除	サーバ管理コマンドの排他情報をリセットして、サーバ管理コマンドを実行できるようにします。	-
cjstopthread	スレッドの削除	指定したスレッドを削除します。 テストモードで実行できます。	更新
cjtracesync	インプロセス HTTP サーバ用トレースファイルの同期	インプロセス HTTP サーバのトレース情報を共有メモリの情報と同期させます。	-
cjunmapsec	ユーザとロールのアンマッピング	ロールからユーザを削除します。	更新

(凡例)

更新：更新系コマンド。J2EE サーバの内容を更新、または構成情報を取得するコマンドです。

参照：参照系コマンド。J2EE サーバの内容を更新しないで、構成状態を表示するだけのコマンドです。

特権：特権系コマンド。J2EE サーバの内容を更新するコマンドで、常にほかのコマンドよりも優先して処理されるコマンドです。

-：該当なし

2. J2EE サーバで使用するコマンド

サーバ管理コマンドの系統ごとの排他制御の条件については、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

cjaddsec (ユーザとロールの追加)

形式

ユーザの場合

```
cjaddsec [<サーバ名称>] [-nameserver <プロバイダURL>]
          -type user -name <ユーザ名> -password <パスワード>
```

ロールの場合

```
cjaddsec [<サーバ名称>] [-nameserver <プロバイダURL>]
          -type role -name <ロール名>
```

機能

ユーザまたはロールを追加します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル, CORBA ネーミングサービスが稼働しているホスト名, およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-type {user|role}

追加するリソースの種別を指定します。

- user : ユーザ
- role : ロール

-name {<ユーザ名>|<ロール名>}

ユーザ名またはロール名を指定します。

-password <パスワード>

ユーザ認証時に使用するパスワードを指定します。

入力例

ユーザの場合

```
cjaddsec MyServer -type user -name scott -password tiger
```

ロールの場合

```
cjaddsec MyServer -type role -name manager
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例: -type <ユーザ名> -name user) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

すでに存在するユーザ名またはロール名は指定できません。

cjdeletesec (ユーザとロールの削除)

形式

```
cjdeletesec [<サーバ名称>] [-nameserver <プロバイダURL>]
             -type {user|role} -name {<ユーザ名>|<ロール名>}
```

2. J2EE サーバで使用するコマンド

機能

ユーザまたはロールを削除します。

削除対象のユーザがロールにマップされている場合、そのロールからアンマップされません。削除対象のロールにマップされているユーザがある場合、そのユーザはアンマップされます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-type {user|role}

削除するリソースの種別を指定します。

- user : ユーザ
- role : ロール

-name {<ユーザ名>|<ロール名>}

ユーザ名またはロール名を指定します。

入力例

ユーザの場合

```
cjdeletesec MyServer -type user -name scott
```

ロールの場合

```
cjdeletesec MyServer -type role -name manager
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

- 2 :
 排他エラーによってコマンドを実行できません。
- 3 :
 タイムアウトが発生しました。
- 9 :
 管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 ,
 Windows Server 2008 x64 ,Windows Server 2008 R2 ,Windows 7 または Windows
 Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) であれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例 : -type <ユーザ名> -name user) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjlistsec (ユーザとロールの一覧表示)

形式

```
cjlistsec [<サーバ名称>] [-nameserver <プロバイダURL>]
          -type {user|role} [-name {<ユーザ名>|<ロール名>}]
```

機能

ユーザまたはロールの一覧を標準出力に出力します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください

2. J2EE サーバで使用するコマンド

さい。

`-type {user|role}`

表示するリソースの種別を指定します。

- user : ユーザ
- role : ロール

`-name {< ユーザ名 >|< ロール名 >}`

ユーザ名またはロール名を指定します。

入力例

すべてのユーザの場合

```
cjlistsec MyServer -type user
```

任意のユーザのロールの場合

```
cjlistsec MyServer -type user -name scott
```

すべてのロールの場合

```
cjlistsec MyServer -type role
```

任意のロールのユーザの場合

```
cjlistsec MyServer -type role -name manager
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : < プロバイダ URL > -nameserver) やオプション名と値を非対応にすること (例 : -type < ユーザ名 > -name user) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称

と大文字・小文字を一致させた文字列を指定する必要があります。
 一覧は大文字小文字の区別なしで、昇順でソートされます。

cjlistthread (スレッド情報の表示)

形式

スタックトレースを出力しない場合

```
cjlistthread [<サーバ名称>] [-nameserver <プロバイダURL>]
```

スタックトレースを出力する場合

```
cjlistthread [<サーバ名称>] [-nameserver <プロバイダURL>] -detail
```

機能

処理中のスレッドに関する、スレッド ID、ルート AP 情報、スレッドの状態、アプリケーション名、実行中メソッドの開始時刻、タイムアウト設定値などの情報を出力します。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

```
<プロトコル名称>::<ホスト名称>:<ポート番号>
```

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-detail

このオプションが指定されている場合、最後に取得されたスタックトレースの情報が出力されます。ただし、スレッドの状態が「running」のスレッドに対してこのオプションを指定してもスタックトレースの情報は出力されません。

入力例

スタックトレースを出力しない場合

```
cjlistthread MyServer
```

スタックトレースを出力する場合

2. J2EE サーバで使用するコマンド

```
cjlistthread MyServer -detail
```

出力形式

```
ThreadID=<スレッドID>,RootApInfo=<ルートAP情報>,Status=<スレッドの状態>,AppName=<アプリケーション名>,StartTime=<開始時刻>,TimeOut=<タイムアウト時間>
```

出力例

スタックトレースを出力しない場合

```
Current Time=HH:MM:SS  
  
ThreadID=aa..aa,RootApInfo=bb..bb,Status=cc..cc,AppName=dd..dd,StartTime=ee..ee,TimeOut=ff..ff  
ThreadID=aa..aa,RootApInfo=bb..bb,Status=cc..cc,AppName=dd..dd,StartTime=ee..ee,TimeOut=ff..ff  
...
```

スタックトレースを出力する場合

```
Current Time=HH:MM:SS  
ThreadID=aa..aa,RootApInfo=bb..bb,Status=cc..cc,AppName=dd..dd,StartTime=ee..ee,TimeOut=ff..ff  
    com.hitachi.XXXX  
        at com.hitachi.YYYY  
        at user.code.UserClass1  
        at com.hitachi.ZZZZ  
...  
...  
ThreadID=aa..aa,RootApInfo=bb..bb,Status=cc..cc,AppName=dd..dd,StartTime=ee..ee,TimeOut=ff..ff  
    com.hitachi.xxxx  
        at com.hitachi.yyyy  
        at user.code.UserClass2  
        at com.hitachi.zzzz  
...  
...
```

(凡例)

aa..aa : スレッドID
bb..bb : ルートAP情報
cc..cc : スレッドの状態
dd..dd : アプリケーション名
ee..ee : 開始時刻
ff..ff : タイムアウト時間

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

3 :

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例: -nameserver <アプリケーション名> -name <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjlisttrn (稼働中の J2EE サーバのトランザクション情報の表示)

形式

```
cjlisttrn [<サーバ名称>] [-nameserver <プロバイダURL>]
          [-gid <グローバルトランザクションID>] [-pending]
          [-time <経過時間>] [-bqual]
```

機能

J2EE サーバで稼働しているトランザクションの状態、経過時間などを一覧表示します。

同一 OS 上で複数の J2EE サーバを起動している場合は、トランザクションに関連するすべての J2EE サーバプロセスごとにこのコマンドを発行する必要があります。

usrconf.properties (J2EE サーバ用ユーザプロパティファイル) の設定が、次の場合に実行できます。

- ejbserver.server.j2ee.feature キーの値が、1.4 の場合
- ejbserver.distributedtx.XATransaction.enabled キーの値が、true の場合

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

出力形式

[Global transaction information]			
Status	GlobalTransactionID	ElapsedTime	BranchType

<トランザクションの状態>	<グローバルトランザクションID>	<経過時間>	<ブランチの種類>
[<XidのブランチID一覧>]			
<トランザクションの状態>	<グローバルトランザクションID>	<経過時間>	<ブランチの種類>
[<XidのブランチID一覧>]			
...			
total count	<表示エントリ数>		

<トランザクションの状態>

Active	: アクティブな状態。
MarkedRollback	: ロールバックにマークされた状態。
Preparing	: プリペア処理中。
Prepared	: Superiorからの決着指示待ち状態。
Committing	: コミット処理中。
RollingBack	: ロールバック処理中。
HeuristicCommit	: 強制コミットされた状態。
HeuristicRollback	: 強制ロールバックされた状態。
HeuristicMixed	: 部分的にコミットまたはロールバックされたため、データの一貫性が失われた状態。
HeuristicHazard	: コミットしたかロールバックしたか、わからない状態。
Invalid	: 上記以外で内部矛盾が発生した状態。

<グローバルトランザクション ID>

対象のグローバルトランザクション ID が表示されます。

<経過時間>

トランザクションを開始してからの経過時間が表示されます。ただし、<ブランチの種類>が「Top (recovered)」または「Sub (recovered)」のトランザクションの場合は、ステータスファイルから回復されてからの経過時間を、「Xid (recovered)」のトランザクションの場合は、Xid 回復後に新規に作成されてからの経過時間が表示されます。

<ブランチの種類>

Top	: トップルートブランチ。
Sub	: サブオーディネートブランチ。
Top (recovered)	: ステータスファイルから回復されたトップルートブランチ。
Sub (recovered)	: ステータスファイルから回復されたサブオーディネートブランチ。
Xid (recovered)	: XAResourceからXidが回復されたが、これに対応するブランチが存在しないため、新規に生成されたブランチ。

<Xidのブランチ ID 一覧>

トランザクションに関連しているすべての Xid のブランチ ID が表示されます。

<表示エントリ数>

表示したエントリ数が表示されます。

戻り値

0:

正常終了しました。

1:

異常終了しました。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例: -nameserver <グローバルトランザクション ID> -gid <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

-gid オプション、-pending オプション、および -time オプションを同時に指定した場合、すべての条件を満たすトランザクションについての情報が出力されます。

cjlisttrnfile (停止中の J2EE サーバのトランザクション情報の表示)

形式

```
cjlisttrnfile [<サーバ名称>] [-gid <グローバルトランザクションID>]
               [-bqual]
```

機能

停止中の J2EE サーバに生成されたステータスファイルからトランザクションの状態などを一覧表示します。コマンドを実行するホストに存在する J2EE サーバだけがコマンドの対象となります。

usrconf.properties (J2EE サーバ用ユーザプロパティファイル) に次のように設定され、J2EE サーバが起動されたことがある場合に実行できます。

2. J2EE サーバで使用するコマンド

<サーバ名称>

上記ステータスファイルを生成した J2EE サーバ名 (ステータスファイルに書き込まれている J2EE サーバ名) が表示されます。

<グローバルトランザクション ID>

対象のグローバルトランザクション ID が表示されます。

<トランザクションの状態>

Prepared	: Superiorからの決着指示待ち状態。
Committing	: コミット処理中。
HeuristicCommit	: 強制コミットされた状態。
HeuristicRollback	: 強制ロールバックされた状態。
HeuristicMixed	: 部分的にコミットまたはロールバックされたため、データの一貫性が失われた状態。
HeuristicHazard	: コミットしたかロールバックしたか、わからない状態。
Invalid	: 上記以外で内部矛盾が発生した状態。

<ブランチの種類>

Top	: トップルートブランチ。
Sub	: サブオーディネートブランチ。

<Xid のブランチ ID 一覧>

トランザクションに関連しているすべての Xid のブランチ ID が表示されます。

<表示エントリ数>

表示したエントリの数が表示されます。

戻り値

0:

正常終了しました (トランザクションが存在しない)。

1:

異常終了しました。

6:

警告終了しました (トランザクションが存在する)。

7:

警告終了しました (トランザクションが存在するかどうかわからない)。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合)。

注意事項

別のホストに存在する J2EE サーバはコマンドの実行対象にできません。

J2EE サーバが起動された状態で、トランザクションの処理中にこのコマンドを実行した場合、表示内容の正当性は保証されません。必ず J2EE サーバが停止している状態でコマンドを実行してください。また、コマンド実行中に J2EE サーバを起動すると、インプロセストランザクションサービスの初期化に失敗することがあります。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<グローバルトランザクション ID> -gid ）やオプション名と値を非対応にすること（例：-gid -bqual <グローバルトランザクション ID>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjmapsec (ユーザとロールのマッピング)

形式

```
cjmapsec [<サーバ名称>] [-nameserver <プロバイダURL>]
        -role <ロール名> -user <ユーザ名> [-user <ユーザ名> ...]
```

機能

ロールにユーザを追加します。一度に複数のユーザのマッピングもできます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-role <ロール名>

ロール名を指定します。

-user <ユーザ名>

ユーザ名を指定します。

2. J2EE サーバで使用するコマンド

入力例

```
cjmapsec MyServer -role manager -user scott
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例: <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例: -role <ユーザ名> -user <ロール名>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

すでに指定したロール名とユーザ名でマッピングされている場合、マッピングはできません。

一度に複数のユーザをマッピングした場合、マッピングできないユーザがあったときでも、すべてのユーザに対してマッピングを試みます。

cjresetsv (サーバ管理コマンドの排他強制解除)

形式

```
cjresetsv [<サーバ名称>] [-nameserver <プロバイダURL>]
```

機能

サーバ管理コマンドの異常終了などによって、サーバ管理コマンドの排他情報に矛盾が

生じる場合があります。サーバ管理コマンドの排他情報に矛盾が生じた場合、ほかにサーバ管理コマンドを実行していなくても、次のメッセージが表示され、サーバ管理コマンドを実行できなくなります。

```
"KDJE37057-E Another command is already running."
```

この場合に、このコマンドを実行すると、サーバ管理コマンドの排他情報をリセットして、サーバ管理コマンドを実行できるようになります。排他制御の対象となるコマンドは、サーバ管理コマンドと `cjstopsv` コマンドです。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

```
<プロトコル名称>::<ホスト名称>:<ポート番号>
```

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

入力例

```
cjresetsv MyServer
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

3:

タイムアウトが発生しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

このコマンドは、コマンドの処理が J2EE サーバ側で実行中でない場合にだけ排他情報を削除します。コマンドの処理が実行中のため排他情報を削除できなかった場合は、エラーメッセージ KDJE42355-W を出力し、異常終了します。

コマンド排他情報が存在しない場合、このコマンドは正常終了します。

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。また、オプション名と値の順序の入れ替え（例：<プロバイダ URL> -nameserver）やオプション名と値を非対応にすること（例：-nameserver <サーバ名称>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjstopthread (スレッドの削除)

形式

```
cjstopthread [<サーバ名称>] [-nameserver <プロバイダURL>]
               -tid <スレッドID> [-tid <スレッドID> ...]
```

機能

指定したスレッドを停止します。テストモードで開始しているアプリケーション中のスレッドも操作できます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-tid <スレッド ID>

停止させるスレッドのスレッド ID を指定します。このオプションは複数回指定できます。なお、スレッド ID の取得方法については、「cjlistthread (スレッド情報の表

示)」を参照してください。

入力例

```
cjstopthread MyServer -tid 100234
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります。そのほかの引数は、サーバ名称より後ろ (サーバ名称を省略した場合はコマンド名の後ろ) にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <プロバイダ URL> -nameserver) やオプション名と値を非対応にすること (例 : -nameserver <スレッド ID> -tid <プロバイダ URL>) はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjtracesync (インプロセス HTTP サーバ用トレースファイルの同期)

形式

```
cjtracesync [-h] [-thr|-comm] <サーバ名称>
```

機能

インプロセス HTTP サーバ使用時に、スレッドトレースおよび通信トレースの情報を共有メモリの情報と同期させます。インプロセス HTTP サーバを使用している場合に、Management Server の機能を利用しないで障害情報を取得するときには、情報を取得す

2. J2EE サーバで使用するコマンド

る直前に、このコマンドを実行してください。

引数

-h

コマンドの使用方法が表示されます。

-thr

スレッドトレース情報を取得する場合に指定します。-thr オプション、-comm オプションのどちらも指定されていない場合、スレッドトレースおよび通信トレースの両方の情報を更新します。

-comm

通信トレース情報を取得する場合に指定します。-thr オプション、-comm オプションのどちらも指定されていない場合、スレッドトレースおよび通信トレースの両方の情報を更新します。

<サーバ名称>

トレース情報を取得する J2EE サーバのサーバ名称を指定します。サーバ名称は必ず指定してください。

戻り値

0:

正常終了しました。

1:

異常終了しました。

9:

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)。

cjunmapsec (ユーザとロールのアンマッピング)

形式

```
cjunmapsec [<サーバ名称>] [-nameserver <プロバイダURL>]
             -role <ロール名> -user <ユーザ名> [-user <ユーザ名> ...]
```

機能

ロールからユーザを削除します。一度に複数のユーザのアンマップもできます。

引数

<サーバ名称>

接続先 J2EE サーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nameserver <プロバイダ URL>

CORBA ネーミングサービスへのアクセスプロトコル、CORBA ネーミングサービスが稼働しているホスト名、およびそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>::<ホスト名称>:<ポート番号>

指定内容の詳細については、「2.1(2) プロバイダ URL について」を参照してください。

-role <ロール名>

削除先のロール名を指定します。

-user <ユーザ名>

削除対象のユーザ名を指定します。

入力例

```
cjunmapsec MyServer -role manager -user scott
```

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

3 :

タイムアウトが発生しました。

9 :

管理者特権がないため、コマンドが実行できません (Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows 7 または Windows Vista の場合)

注意事項

コマンドの引数にサーバ名称を指定する場合、コマンド名の直後にする必要があります

2. J2EE サーバで使用するコマンド

す。そのほかの引数は、サーバ名称より後ろ（サーバ名称を省略した場合はコマンド名の後ろ）にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<プロバイダ URL> -nameserver）やオプション名と値を非対応にすること（例：-role <ユーザ名> -user <ロール名>）はできません。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

一度に複数のユーザをアンマップした場合、アンマップできないユーザがあったときでも、すべてのユーザに対してアンマップを試みます。

3

バッチサーバで使用するコマンド

この章では、バッチサーバで使用するコマンドの入力形式、機能などについて説明します。

3.1 バッチサーバで使用するコマンドの概要

3.2 バッチサーバを操作するコマンド

3.3 バッチアプリケーションで使用するコマンド

3.4 バッチサーバで使用するリソース操作コマンド

3.5 バッチサーバで使用するそのほかのコマンド

3.1 バッチサーバで使用するコマンドの概要

ここでは、バッチサーバで使用するコマンドの種類、プロバイダ URL について説明します。

(1) バッチサーバで使用するコマンドの種類

バッチサーバで使用するコマンドの種類について説明します。

バッチサーバで使用するコマンドの種類と格納先を、次の表に示します。

表 3-1 バッチサーバで使用するコマンドの種類と格納先

バッチサーバで使用するコマンドの種類	コマンドの格納先
バッチサーバを操作するコマンド	<ul style="list-style-type: none"> Windows の場合 <Cosminexus のインストールディレクトリ>¥CC¥server¥bin¥ UNIX の場合 /opt/Cosminexus/CC/server/bin/
バッチアプリケーションで使用するコマンド	<ul style="list-style-type: none"> Windows の場合 <Cosminexus のインストールディレクトリ>¥CC¥batch¥bin¥ UNIX の場合 /opt/Cosminexus/CC/batch/bin/
バッチサーバで使用するリソース操作コマンド	<ul style="list-style-type: none"> Windows の場合 <Cosminexus のインストールディレクトリ>¥CC¥admin¥bin¥ UNIX の場合 /opt/Cosminexus/CC/admin/bin/
バッチサーバで使用するそのほかコマンド	

(2) プロバイダ URL について

CORBA ネーミングサービスへのアクセスプロトコル、および CORBA ネーミングサービスが稼働しているホスト名称、ならびにそれが使用しているポート番号を次に示す形式で指定します。

<プロトコル名称>.:.<ホスト名称>.<ポート番号>

なお、該当する CORBA ネーミングサービスは、接続先バッチサーバがネーミングサービスとして利用しているものです。サポートしているアクセスプロトコルは corbaname だけです。

<ホスト名称> には CORBA ネーミングサービスが稼働しているホスト名称を指定します。ただし、接続先バッチサーバの usrconf.properties (バッチサーバ用ユーザプロパティファイル) で vbroker.se.iiop_tp.host キーに値を指定している場合は、vbroker.se.iiop_tp.host キーに指定した値を <ホスト名称> に指定します。

<ホスト名称> を省略した場合は次のようになります。

usrconf.properties (サーバ管理コマンド用システムプロパティファイル) の
"ejbserver.naming.protocol" キーの値をアクセスプロトコルとして使用します。

"ejbserver.naming.host" キーの値をホスト名称として使用します。

"ejbserver.naming.port" キーの値をポート番号として使用します。

また、<ホスト名称> のデフォルト値は次のとおりです。

"ejbserver.naming.protocol" キー : corbaname

"ejbserver.naming.host" キー : localhost

"ejbserver.naming.port" キー : 900

3.2 バッチサーバを操作するコマンド

ここでは、バッチサーバを操作するコマンドについて説明します。

バッチサーバを操作するコマンドの一覧を、次の表に示します。

表 3-2 バッチサーバを操作するコマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjdumps ¹	バッチサーバのスレッドダンプの取得	バッチサーバで障害が発生した場合などに、JavaVMのスレッドダンプを取得します。	-
cjenvsetup ¹	Component Container 管理者のセットアップ	インストールファイルの属性を変更して、root 権限を持つユーザ以外で運用できるようにします。このコマンドは、UNIX 用です。	-
cjenvupdate ¹	作業ディレクトリおよびユーザ定義ファイルの移行	Cosminexus Component Container の旧環境で利用されていた作業ディレクトリおよびユーザ定義ファイルを新環境で動作する形式に移行します。	-
cjgetsysinfo ¹	OS 状態情報の取得	OS の状態情報を取得します。	-
cjrarupdate ¹	リソースアダプタのバージョンアップ	Cosminexus Component Container にインポート・デプロイされているリソースアダプタをバージョンアップします。	-
cjsetup	バッチサーバのセットアップとアンセットアップ	バッチサーバの環境をセットアップして、デフォルトのユーザ定義を作成します。	-
cjsleep ¹	バッチサーバの起動間隔	指定された秒数だけ実行を一時停止します。バッチファイルなどでサーバの起動間隔を空けるために使用します。このコマンドは、Windows 用です。	-
cjstartsv	バッチサーバの開始	バッチサーバを開始します。	-
cjstopsv ¹	バッチサーバの停止	バッチサーバを停止します。	更新 ²

(凡例)

更新：更新系コマンド。バッチサーバの内容を更新、または構成情報を取得するコマンドです。

-：該当なし

注 1

コマンドの詳細は、「2.2 J2EE サーバを操作するコマンド」を参照してください。なお、参照する場合は「J2EE サーバ」を「バッチサーバ」と読み替えてください。

注 2

-f オプションを指定した場合は、-（該当なし）になります。

サーバ管理コマンドの系統ごとの排他制御の条件については、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

セットアップ系コマンド (cjenvsetup, cjenvupdate, cjrarupdate, cjsetup) は単独で実行してください。ほかのセットアップ系コマンドと同時に実行しないでください。

cjsetup (バッチサーバのセットアップとアンセットアップ)

形式

セットアップの場合

```
cjsetup -batch [<サーバ名称>]
```

アンセットアップの場合

```
cjsetup -d <サーバ名称>
```

機能

指定したサーバ名称のバッチサーバの環境をセットアップします。

次のディレクトリにセットアップしたバッチサーバ用のデフォルトのユーザ定義を作成します。

- Windows の場合
<Cosminexus のインストールディレクトリ >¥CC¥server¥usrconf¥ejb¥< サーバ名称 >¥
- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称 >/

なお、-d オプションを指定した場合、次に示すバッチサーバの環境 (ディレクトリおよびファイル) を削除します。

ユーザ定義ファイルの格納用ディレクトリ以下

- Windows の場合
<Cosminexus のインストールディレクトリ >¥CC¥server¥usrconf¥ejb¥< サーバ名称 >¥
- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称 >/

一時作業用ディレクトリ以下

- Windows の場合
<Cosminexus のインストールディレクトリ >¥CC¥server¥repository¥< サーバ名称 >¥
- UNIX の場合
/opt/Cosminexus/CC/server/repository/< サーバ名称 >/

3. バッチサーバで使用するコマンド

作業用ディレクトリ以下 (デフォルト)

- Windows の場合
 <Cosminexus のインストールディレクトリ >%CC%¥server¥public¥ejb¥< サーバ名
 称 >¥
- UNIX の場合
 /opt/Cosminexus/CC/server/public/ejb/< サーバ名称 >/

なお、作業用ディレクトリについては、ユーザ定義ファイル (usrconf.cfg) の ejb.public.directory キーにデフォルト値以外が指定されていた場合、そのディレクトリ以下も削除します。

引数

<サーバ名称 >

セットアップするバッチサーバのサーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。ただし、アンセットアップ時は省略できません。また、ホスト名称にサーバ名称として使用できない文字が含まれている場合はエラーになります。

半角英数字 (0 ~ 9, A ~ Z, a ~ z), アンダースコア (_) またはハイフン (-) を使用できます。サーバ名称の先頭の文字は英数字でなければなりません。また、サーバ名称は、255 文字以下、かつマニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」のバッチサーバの作業ディレクトリの説明を参照して見積もった範囲内の長さで指定してください。

-batch

バッチサーバの環境を構築する場合に指定します。

-d

バッチサーバの環境を削除する場合に指定します。サーバ名称は省略できません。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

注意事項

UNIX の場合、cjsetup コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

UNIX の場合にアンセットアップ (cjsetup -d <サーバ名称 >) を実行すると、削除対象のディレクトリ、ファイルのアクセス権の設定および使用状態に関係なく、すべて

削除します。

セットアップ済みのサーバ名称を指定して再セットアップ (`cjsetup -batch <サーバ名称>`) を行うと、エラーとなります。

Windows の場合、サーバ名称の大文字・小文字は区別されません。すでに作成済みのサーバ名称と、大文字・小文字の区別以外は一致している文字列を `cjsetup` のサーバ名称として指定した場合、次のように動作します。

- セットアップの場合 (`cjsetup -batch <サーバ名称>`)
セットアップに失敗します。
- アンセットアップの場合 (`cjsetup -d <サーバ名称>`)
すでに作成済みである、`<サーバ名称>` と大文字・小文字の区別以外は一致しているサーバが削除されます。

サーバ管理コマンド、バッチサーバの開始または停止コマンドを実行する場合、これらのコマンドにサーバ名称を引数として指定するときは、`cjsetup` コマンドで作成したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

起動中のバッチサーバに対して、アンセットアップ (`cjsetup -d <サーバ名称>`) を実行しないでください。

Windows の場合、削除対象のファイルの属性が読み取り専用、または使用中のときに、アンセットアップ (`cjsetup -d <サーバ名称>`) を実行しないでください。

cjstartsv (バッチサーバの開始)

形式

```
cjstartsv [<サーバ名称>] [-nosecurity]
```

機能

指定したサーバ名称のバッチサーバを開始します。サーバ名称を省略したときは、ホスト名称と同じサーバ名称のバッチサーバを開始します。

引数

<サーバ名称>

開始するバッチサーバのサーバ名称を指定します。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-nosecurity

SecurityManager を解除するために指定します。

ユーザ定義ファイル

Windows の場合

- <Cosminexus のインストールディレクトリ>\%CC%\server\%usrconf%\ejb\<サーバ

3. バッチサーバで使用するコマンド

名称 >¥usrconf.cfg

- <Cosminexus のインストールディレクトリ >¥CC¥server¥usrconf¥ejb¥< サーバ名称 >¥usrconf.properties

UNIX の場合

- /opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称 >/usrconf.cfg
- /opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称 >/usrconf.properties

戻り値

0 :

正常終了しました。

1 以上 (143 以外) :

異常終了しました。

143 :

コマンド実行中に次の操作をしました (Windows の場合)。

- コマンドを起動しているコマンドプロンプトのウィンドウを「 × 」ボタンを押下して閉じた。
- Windows システムをシャットダウンした。
- Windows システムをログオフした。

注意事項

バッチサーバを自動起動するように設定できます。バッチサーバの起動については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 拡張編」を参照してください。

Windows の場合、cjstartsv コマンドの実行には、Administrator 権限が必要です。

UNIX の場合、cjstartsv コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

コマンドの引数にサーバ名称を指定する場合、cjsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

3.3 バッチアプリケーションで使用するコマンド

ここでは、バッチアプリケーションで使用するコマンドについて説明します。

バッチアプリケーションで使用するコマンドの一覧を、次の表に示します。

表 3-3 バッチアプリケーションで使用するコマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjexecjob	バッチアプリケーションの実行	バッチアプリケーションを実行します。	-
ckilljob	バッチアプリケーションの停止またはバッチサーバの強制停止	バッチアプリケーションを停止します。バッチアプリケーションの停止に失敗した場合、バッチサーバを強制停止します。	-
cjlistjob	バッチアプリケーション情報の一覧表示	バッチアプリケーションの情報を標準出力に出力します。	-

(凡例)

- : 該当なし

上記の表のコマンドは、バッチアプリケーションのスケジューリング機能を使用できません。バッチアプリケーションのスケジューリング機能については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 拡張編」を参照してください。

以降、ここでは、バッチアプリケーションのスケジューリング機能をスケジューリング機能といいます。

サーバ管理コマンドの系統ごとの排他制御の条件については、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

cjexecjob (バッチアプリケーションの実行)

形式

スケジューリング機能を使用しない場合

```
cjexecjob <サーバ名称> [[<Javaオプション>]...]
<Javaアプリケーションクラス名>
[[<mainメソッドに渡す引数>]...]
```

スケジューリング機能を使用する場合

```
cjexecjob [<スケジューラグループ名>] [-jobID <ジョブID>]
[[<Javaオプション>]...]
<Javaアプリケーションクラス名>
[[<mainメソッドに渡す引数>]...]
```

3. バッチサーバで使用するコマンド

機能

スケジューリング機能を使用しない場合は、指定したサーバ名称のバッチサーバでバッチアプリケーションを実行します。

スケジューリング機能を使用する場合は、CTM によって振り分けられたバッチサーバでバッチアプリケーションを実行します。なお、ジョブが実行できるバッチサーバがないときは、バッチアプリケーションをスケジュールキューに待機させます。

<Java オプション> 以降の引数には最大 1024 バイトまで指定できます。

cjexecjob コマンドを実行した場合、および cjexecjob コマンドの処理が終了した場合は、メッセージが出力されます。cjexecjob コマンドが出力するメッセージは、usrconf.cfg (バッチアプリケーション用オプション定義ファイル) の batch.log.stdout.enabled キーによって抑止できます。

cjexecjob コマンドのオプションは、usrconf.cfg (バッチアプリケーション用オプション定義ファイル) で指定します。また、システムプロパティは、usrconf.properties (バッチアプリケーション用ユーザプロパティファイル) で指定します。usrconf.cfg (バッチアプリケーション用オプション定義ファイル) および usrconf.properties (バッチアプリケーション用ユーザプロパティファイル) が格納されているディレクトリは、次の順で検索されます。

1. 環境変数 (CJBATCHUSRCONFDIR) に指定したディレクトリ
2. cjexecjob コマンドを実行したディレクトリ

注 環境変数 (CJBATCHUSRCONFDIR) にはディレクトリを絶対パスで指定してください。

同じキーのシステムプロパティを複数指定した場合、最後に指定したシステムプロパティが優先されます。

同じキーのシステムプロパティを異なる手段で指定した場合の優先順位は次のとおりです。

1. cjexecjob コマンドの引数
2. usrconf.properties (バッチアプリケーション用ユーザプロパティファイル)
3. usrconf.cfg (バッチアプリケーション用オプション定義ファイル)
4. usrconf.properties (バッチサーバ用ユーザプロパティファイル)
5. usrconf.cfg (バッチサーバ用オプション定義ファイル)

引数

<サーバ名称>

バッチアプリケーションを実行するバッチサーバの名称を指定します。

<スケジュールグループ名>

CTM がバッチアプリケーションの実行をスケジューリングするときに割り当てるバッチサーバのグループ名を指定します。

スケジューリンググループ名は、usrconf.cfg (バッチアプリケーション用オプション定義ファイル) でも指定できます。スケジューリンググループ名の指定の優先順位は次のとおりです。

- cjexecjob コマンドの引数
- usrconf.cfg (バッチアプリケーション用オプション定義ファイル)

なお、スケジューリンググループ名の指定は省略できます。スケジューリンググループ名の指定を省略した場合のデフォルト値は "JOBGROUP" です。

-jobID <ジョブ ID>

実行するバッチアプリケーションのジョブ ID をユニークな 1 ~ 32 文字の英数字で指定します。

-jobID <ジョブ ID> を省略した場合、ジョブ ID は自動生成されます。ジョブ ID の自動生成については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 拡張編」のバッチアプリケーションのスケジューリングの説明を参照してください。

<Java オプション>

JavaVM の起動オプションを指定します。

指定できるオプションを次に示します。

- -cp
- -classpath
- -D<name>=<value>

<Java アプリケーションクラス名>

パッケージ名を含めた Java アプリケーションのクラス名を指定します。

<main メソッドに渡す引数>

Java アプリケーションの main メソッドに渡す引数を指定します。

入力例

スケジューリング機能を使用しない場合

```
cjexecjob MyBatchServer com.hitachi.mypackage.MyBatchApp
```

スケジューリング機能を使用する場合

```
cjexecjob JOBGROUP com.hitachi.mypackage.MyBatchApp
```

戻り値

<public static int main(String[]) メソッドから return した値> :

public static int main(String[]) メソッドを使用した場合に返却されます。

<System.exit(),Runtime.halt(), Runtime.exit() の引数に指定した値> :

System.exit(),Runtime.halt(), Runtime.exit() を使用した場合に返却されます。

3. バッチサーバで使用するコマンド

0:

`public static void main(String[])` メソッドを使用して、正常終了しました。

1:

バッチアプリケーションの実行に失敗しました。

または、バッチアプリケーションを強制終了しました。

注意事項

UNIX の場合、バッチサーバを起動した Component Container 管理者またはスーパーユーザがコマンドを実行できます。

Windows の場合、Administrator 権限を持つユーザがコマンドを実行できます。

Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合、管理者特権を持つユーザがコマンドを実行できます。

サーバ名称は、バッチサーバのセットアップ時に指定したサーバ名称と大文字・小文字を一致させた文字列を指定してください。

クラスパスを複数指定した場合、すべての指定が有効になります。例えば、"`-cp a ; b`" を指定した場合、"`-cp a ; b`" (Windows の場合) と同等になります。

クラスパスは指定した順に検索します。複数のクラスパスを異なる手段で指定した場合、次の順番で検索します。

1. `usrconf.cfg` (バッチサーバ用オプション定義ファイル)
2. `ejexecjob` コマンドの引数
3. `usrconf.cfg` (バッチアプリケーション用オプション定義ファイル)

クラスパスは `CLASSPATH` 環境変数には指定できません。指定しても無視されます。また、`-cp ; classpath` の指定がない場合、デフォルトのクラスパスとしてカレントディレクトリは使用されません。

クラスパスに相対パスを指定した場合、コマンドを実行したディレクトリからの相対パスとして扱います。

`ejexecjob` コマンドの引数にマルチバイト文字を使用する場合は、次の個所で使用しているマルチバイト文字とエンコードを統一してください。

- `usrconf.cfg` (バッチアプリケーション用オプション定義ファイル)
- `usrconf.cfg` (バッチサーバ用オプション定義ファイル)
- バッチアプリケーションのソースコード (バッチアプリケーション内の `java.lang.System.out`, または `java.lang.System.err`)

スケジューリング機能を使用する場合、`ejexecjob` コマンドとバッチサーバ間の通信タイムアウトの設定を `usrconf.cfg` (バッチアプリケーション用オプション定義ファイル) で指定できます。デフォルト値は 0 で、無限待ちとなります。

cjkilljob (バッチアプリケーションの停止またはバッチサーバの強制停止)

形式

スケジューリング機能を使用しない場合

```
cjkilljob <サーバ名称>
```

スケジューリング機能を使用する場合

```
cjkilljob { <サーバ名称> | [<スケジュールグループ名>] -jobID <ジョブID>
}
```

機能

指定したサーバ名称のバッチサーバ上で実行中のバッチアプリケーションを停止します。実行中のバッチアプリケーションの停止に失敗した場合、バッチサーバを強制停止しします。

スケジューリング機能を使用する場合で、スケジュールグループ名を指定したときは、-jobID で指定したジョブ ID を持つ実行中のバッチアプリケーションを停止します。また、-jobID で指定したジョブ ID を持つバッチアプリケーションが CTM のスケジュールキューで待機中のときは、バッチアプリケーションの削除予約ができます。削除予約したバッチアプリケーションは、スケジュールキューから出たときに CTM によって削除されます。

cjkilljob コマンドのオプションは、usrconf.cfg (バッチアプリケーション用オプション定義ファイル) で指定できます。usrconf.cfg (バッチアプリケーション用オプション定義ファイル) が格納されているディレクトリは、次の順で検索されます。

1. 環境変数 (CJBATCHUSRCONFDIR) に指定したディレクトリ
2. cjkilljob コマンドを実行したディレクトリ

注 環境変数 (CJBATCHUSRCONFDIR) にはディレクトリを絶対パスで指定してください。

引数

<サーバ名称>

停止するバッチアプリケーションを実行中のバッチサーバの名称を指定します。

<スケジュールグループ名>

CTM がバッチアプリケーションの実行をスケジューリングするときに割り当てるバッチサーバのグループ名を指定します。

スケジュールグループ名は、usrconf.cfg (バッチアプリケーション用オプション定義ファイル) でも指定できます。スケジュールグループ名の指定の優先順位は次のとおりです。

3. バッチサーバで使用するコマンド

- cjexecjob コマンドの引数
- usrconf.cfg (バッチアプリケーション用オプション定義ファイル)

なお、スケジュールグループ名の指定は省略できます。スケジュールグループ名の指定を省略した場合のデフォルト値は "JOBGROUP" です。

-jobID <ジョブ ID>

停止するバッチアプリケーションのジョブ ID を指定します。

入力例

サーバ名を指定する場合

```
ckilljob MyBatchServer
```

ジョブ ID を指定する場合

```
ckilljob JOBGROUP -jobID HOGE
```

戻り値

0 :

バッチアプリケーションを停止しました。

または、スケジュールキューに待機していたジョブ ID の削除を予約しました。

1 :

バッチサーバを強制停止しました。

2 :

異常終了しました。

または次のどれかで警告終了をしました。

- バッチアプリケーションが実行されていない。
- 指定したジョブ ID が見つからない。
- すでに削除予約されている。

注意事項

UNIX の場合、バッチサーバを起動した Component Container 管理者またはスーパーユーザがコマンドを実行できます。

Windows の場合、Administrator 権限を持つユーザがコマンドを実行できます。

Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合、管理者特権を持つユーザがコマンドを実行できます。

サーバ名称は、バッチサーバのセットアップ時に指定したサーバ名称と大文字・小文字を一致させた文字列を指定してください。

スケジューリング機能を使用する場合、ckilljob コマンドとバッチサーバ間の通信タイムアウトの設定を usrconf.cfg (バッチアプリケーション用オプション定義ファイ

ル)で指定できます。デフォルト値は0で、無限待ちとなります。

cjlistjob (バッチアプリケーション情報の一覧表示)

形式

スケジューリング機能を使用しない場合

```
cjlistjob <サーバ名称>
```

スケジューリング機能を使用する場合

```
cjlistjob {[ [<スケジュールグループ名>]...} | [-all] }
```

機能

スケジューリング機能を使用しない場合

指定したサーバ名称のバッチサーバで実行中のバッチアプリケーションの情報を標準出力に1行で出力します。

出力するバッチアプリケーションの情報を次に示します。

- バッチアプリケーションの状態
- バッチアプリケーション名
- PRF のルート AP 情報
- cjexecjob コマンドの実行時刻

スケジューリング機能を使用する場合

指定したスケジュールグループで実行中のバッチアプリケーションの情報を標準出力に1行で出力します。

出力するバッチアプリケーションの情報を次に示します。

- スケジュールグループ名
- バッチアプリケーションの状態
- バッチアプリケーション名
- PRF のルート AP 情報
- cjexecjob コマンドの実行時刻
- バッチアプリケーションの待機開始, 実行開始, または強制停止受付時刻
- ジョブ ID
- バッチアプリケーションを実行しているサーバ名

cjlistjob コマンドのオプションは、usrconf.cfg (バッチアプリケーション用オプション定義ファイル) で指定できます。usrconf.cfg (バッチアプリケーション用オプション定義ファイル) が格納されているディレクトリは、次の順で検索されます。

1. 環境変数 (CJBATCHUSRCONFDIR) に指定したディレクトリ
2. cjlistjob コマンドを実行したディレクトリ

注 環境変数 (CJBATCHUSRCONFDIR) にはディレクトリを絶対パスで指定してください。

3. バッチサーバで使用するコマンド

引数

<サーバ名称>

情報を取得したいバッチアプリケーションを実行しているバッチサーバの名称を指定します。

サーバ名称は、バッチサーバのセットアップ時に指定したサーバ名称と大文字・小文字を一致させた文字列を指定してください。

<スケジュールグループ名>

情報を取得したいバッチアプリケーションを実行しているスケジュールグループの名称を指定します。複数のスケジュールグループをコンマ(,)区切りで指定できます。

なお、スケジュールグループ名の指定は省略できます。スケジュールグループ名の指定を省略した場合は、このコマンドの `usrconf.cfg` (バッチアプリケーション用オプション定義ファイル) 内にある `batch.schedule.group.name` が使用されます。フォルト値は "JOBGROUP" です。

-all

同一マシン内のバッチサーバが使用しているすべてのスケジュールグループのバッチアプリケーション情報を表示する場合に指定します。

入力例

スケジューリング機能を使用しない場合

```
cjlistjob MyBacthServer
```

スケジューリング機能を使用する場合

```
cjlistjob JOBGROUP
```

戻り値

0:

バッチアプリケーション情報の一覧を表示しました。

1:

バッチアプリケーション情報の一覧の表示に失敗しました。

出力形式

スケジューリング機能を使用しない場合

<バッチアプリケーションの状態> <バッチアプリケーション名> <PRFのルートAP情報> <cjexecjobコマンドの実行時刻>

スケジューリング機能を使用する場合

<スケジュールグループ名> <バッチアプリケーションの状態> <バッチアプリケーション名> <PRFのルートAP情報> <cjexecjobコマンドの実行時刻> <バッチアプリケーション待機開始, 実行時間, または強制停止受付時間> <ジョブID> <ジョブを実行しているバッチサーバ名>

(凡例) : 半角スペース

次に出力項目について説明します。

<スケジュールグループ名>

バッチアプリケーションの実行リクエストが振り分けられるスケジュールグループの名称を表示します。

<バッチアプリケーションの状態>

バッチアプリケーションの次の状態を表示します。

- バッチアプリケーション実行中: running
- バッチアプリケーション待機中: waiting
- バッチアプリケーション強制停止中: forceStopping

<バッチアプリケーション名>

cjexecjob コマンドで指定した Java アプリケーションのクラス名を表示します。

<PRF のルート AP 情報>

PRF のルートアプリケーションの通信番号を, 先頭けたが 0x である 16 けたの 16 進数で表示します。

<cjexecjob コマンドの実行時刻>

cjexecjob コマンドを実行した時刻を次の形式で表示します。

yyyy/mm/dd hh:mm:ss.ssssss

(凡例) : 半角スペース yyyy/mm/dd hh:mm:ss.ssssss : 西暦年 / 月 / 日 時間 / 分 / 秒

<バッチアプリケーションの待機開始, 実行開始, または強制停止受付時刻>

スケジューリング機能を使用する場合に出力されます。

バッチアプリケーションの状態別に, バッチアプリケーションの開始時刻, 待機時刻, または強制停止受付時刻を次の形式で表示します。

yyyy/mm/dd hh:mm:ss.ssssss

(凡例) : 半角スペース yyyy/mm/dd hh:mm:ss.ssssss : 西暦年 / 月 / 日 時間 / 分 / 秒

3. バッチサーバで使用するコマンド

バッチアプリケーションの状態と表示される時刻の対応を次の表に示します。

表 3-4 バッチアプリケーションの状態と表示される時刻

バッチアプリケーションの状態	表示される時刻	説明
waiting	待機開始時刻	バッチアプリケーションがスケジュールキューに入った時刻
running	実行開始時刻	バッチアプリケーションがスケジュールキューから出た時刻
forceStopping	強制停止受付時刻	スケジュールキューのバッチアプリケーションに強制停止予約をした時刻

< ジョブ ID >

バッチアプリケーションのジョブ ID を表示します。

< ジョブを実行しているバッチサーバ名 >

スケジューリング機能を使用する場合に出力されます。

バッチアプリケーションを実行しているバッチサーバ名を表示します。バッチアプリケーションが待機中の場合は、「-」を表示します。

出力例

指定したサーバ名称のバッチサーバの情報

```
running com.hitachi.mypackage.batchApp1 0x0000000000123456 2008/04/14
17:27:35.689012
```

指定したスケジュールグループの情報

```
JOBGROUP running com.hitachi.mypackage.batchApp1 0x0000000000123456 2008/
04/14 17:27:35.689012 2008/04/14 17:27:37.182777 HOGE MybatchServer1
JOBGROUP running com.hitachi.mypackage.batchApp2 0x0000000002345678 2008/
04/14 17:45:20.123456 2008/04/14 19:21:56.271354 102 MybatchServer2
JOBGROUP running com.hitachi.mypackage.batchApp3 0x0000000034567890 2008/
04/14 18:15:54.397890 2008/04/14 19:00:00.123447 #5HL390_G3CV7
MybatchServer3
JOBGROUP waiting com.hitachi.mypackage.batchApp4 0x0000000045678901 2008/
04/14 18:30:24.125444 2008/04/14 18:30:25.006220 112345 -
```

(凡例) : 半角スペース

注意事項

UNIX の場合、バッチサーバを起動した Component Container 管理者またはスーパーユーザがコマンドを実行できます。

Windows の場合、Administrator 権限を持つユーザがコマンドを実行できます。

Windows Server 2008 x86、Windows Server 2008 x64、Windows Server 2008 R2、Windows 7 または Windows Vista の場合、管理者特権を持つユーザがコマンドを実行できます。

バッチアプリケーションがない場合、情報は何も出力されないでコマンドが正常終了します。

スケジューリング機能を使用する場合、バッチアプリケーション情報の一覧表示のタイムアウト設定を `usrconf.cfg` (バッチアプリケーション用オプション定義ファイル) で指定できます。スケジューリング機能を使用する場合の設定については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 拡張編」を参照してください。

スケジューリング機能を使用する場合、`cjlistjob` コマンドとバッチサーバ間の通信タイムアウトの設定を `usrconf.cfg` (バッチアプリケーション用オプション定義ファイル) で指定できます。デフォルト値は 0 で、無限待ちとなります。

3.4 バッチサーバで使用するリソース操作コマンド

ここでは、バッチサーバで使用するリソース操作コマンドについて説明します。

コマンドの詳細は、「2.4 J2EE サーバで使用するリソース操作コマンド」を参照してください。なお、参照する場合は「J2EE サーバ」を「バッチサーバ」と読み替えてください。

バッチサーバで使用するリソース操作コマンドの一覧を、次の表に示します。

表 3-5 バッチサーバで使用するリソース操作コマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjclearpool	コネクションプール内のコネクション削除	リソースアダプタのコネクションをクリアします。	更新
cjcopyres	リソースのコピー	RAR ファイルのプロパティをコピーします。	更新
cjdeleteres	リソースの削除	インポート済みの RAR ファイルを削除します。	更新
cjdeployrar	リソースアダプタのデプロイ	リソースアダプタをデプロイします。	更新
cjexportrar	リソースアダプタのエクスポート	バッチサーバ上のリソースアダプタをエクスポートします。	更新
cjgetrarprop	RAR ファイルの属性の取得	デプロイ済みの RAR ファイルの属性を取得して、属性ファイルを生成します。	更新
cjgetresprop	リソースの属性の取得	デプロイ前の RAR ファイルに含まれるリソースの属性を取得して、属性ファイルを生成します。	更新
cjimportres	リソースのインポート	RAR ファイルをインポートします。	更新
cjlistpool	コネクションプールの一覧表示	指定されたリソースアダプタ、またはすべてのリソースアダプタのコネクションプール状態を標準出力に出力します。	参照
cjlistrar	リソースアダプタの一覧表示	デプロイされているすべてのリソースアダプタの一覧を標準出力に出力します。	参照
cjlistres	リソースの一覧表示	インポート済みの RAR ファイルの一覧を標準出力に出力します。	参照
cjresumepool	メンバコネクションプールの再開	クラスタコネクションプールのメンバコネクションプールの再開します。	更新

コマンド名称	分類	概要	コマンドの系統
cjsetrarprop	RAR 属性設定	デプロイ済みの RAR ファイルに含まれるリソースの属性を属性ファイルに指定された値に変更します。	更新
cjsetresprop	リソースの属性設定	デプロイ前の RAR ファイルに含まれるリソースの属性を、属性ファイルに指定された値に変更します。	更新
cjstartrar	リソースアダプタの開始	リソースアダプタを開始します。	更新
cjstoprar	リソースアダプタの停止	開始されているリソースアダプタを停止します。	更新
cjsuspendpool	メンバコネクションプールの一時停止	クラスタコネクションプールのメンバコネクションプールを一時停止します。	更新
cjundeployrar	リソースアダプタのアンデプロイ	デプロイされているリソースアダプタをアンデプロイします。	更新

(凡例)

更新：更新系コマンド。バッチサーバの内容を更新，または構成情報を取得するコマンドです。

参照：参照系コマンド。バッチサーバの内容を更新しないで，構成状態を表示するだけのコマンドです。

サーバ管理コマンドの系統ごとの排他制御の条件については，マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

3.5 バッチサーバで使用するそのほかのコマンド

ここでは、バッチサーバで使用するそのほかのコマンドについて説明します。

コマンドの詳細は、「2.5 J2EE サーバで使用するそのほかのコマンド」を参照してください。なお、参照する場合は「J2EE サーバ」を「バッチサーバ」と読み替えてください。

バッチサーバで使用するそのほかのコマンドの一覧を、次の表に示します。

表 3-6 バッチサーバで使用するそのほかのコマンドの一覧

コマンド名称	分類	概要	コマンドの系統
cjresetsv	サーバ管理コマンドの 排他強制解除	サーバ管理コマンドの排他情報をリセットして、サーバ管理コマンドを実行できるようにします。	-

(凡例)

- : 該当なし

サーバ管理コマンドの系統ごとの排他制御の条件については、マニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」を参照してください。

4

性能解析トレースで使用するコマンド

この章では、性能解析トレースで使用するコマンドの入力形式、機能などについて説明します。

4.1 性能解析トレースで使用するコマンドの一覧

4.2 性能解析トレースで使用するコマンドの詳細

4.1 性能解析トレースで使用するコマンドの一覧

性能解析トレースで使用するコマンドは、`cprf` で始まるコマンドです。性能解析トレースで使用するコマンドの一覧を、次の表に示します。

表 4-1 性能解析トレースで使用するコマンドの一覧

コマンド名称	分類	概要
<code>cprfd</code>	性能解析トレース情報の編集出力	性能解析トレース情報を編集出力します。
<code>cprfflush</code>	バッファ内容の強制ファイル出力	バッファ内のトレース情報を出力します。
<code>cprfgetpid</code>	PRF デーモンプロセス ID の取得	PRF デーモンのプロセス ID を取得します。
<code>cprflevel</code>	PRF トレース取得レベルの表示と変更	PRF トレース取得レベルを表示または変更します。
<code>cprfstart</code>	PRF デーモンの開始	PRF デーモン (<code>cprfd</code>) を起動します。
<code>cprfstop</code>	PRF デーモンの停止	PRF デーモン (<code>cprfd</code>) を停止します。

4.2 性能解析トレースで使用するコマンドの詳細

性能解析トレースで使用する各コマンドの入力形式、機能などを次に示します。

コマンドの格納先

性能解析トレースで使用するコマンドは、次のディレクトリに格納されています。

- Windows の場合
`<Cosminexus のインストールディレクトリ>\PRF\bin\`
- UNIX の場合
`/opt/Cosminexus/PRF/bin/`

cprfed (性能解析トレース情報の編集出力)

形式

```
cprfed [-h] | [-Dump] | -CSV] [-Time <開始時刻>, <終了時刻>]
        [-ProcessID <プロセスID>[, <プロセスID>...]]
        [-TraceFile <PRFトレースファイル名> [, <PRFトレースファイル名>...]]
        | -AllTraceFile ]
```

機能

PRF トレース情報を入力して、性能解析トレース情報を編集出力します。

引数

-h

コマンドの使用方法が表示されます。

-Dump

詳細情報をダンプ形式で出力します。

-CSV

詳細情報を CSV 形式で出力します。

-Time < 開始時刻 >, < 終了時刻 >

指定された時刻内に取得されたトレース情報だけを出力します。

開始時刻、終了時刻は、hhmmss[MMdd[yyyy]] 形式で指定します。MMdd, yyyy は省略できます。省略した場合は、コマンドを実行した月日年が指定されます。

時刻の文字列の形式：hhmmss[MMdd[yyyy]]

hh：時 (00 ~ 23)

mm：分 (00 ~ 59)

ss：秒 (00 ~ 59)

MM：月 (01 ~ 12)

4. 性能解析トレースで使用するコマンド

dd : 日 (01 ~ 31)

yyyy : 年 (1970 ~ 20xx)

2003 年 10 月 11 日にコマンドを実行した場合の引数の指定例とその編集範囲の例を次に示します。

引数の指定	編集範囲
-Time 102345,	2003 年 10 月 11 日 10 時 23 分 45 秒からトレースファイルの最後まで。
-Time ,102345	トレースファイルの最初から 2003 年 10 月 11 日 10 時 23 分 45 秒まで。
-Time 102345,102350	2003 年 10 月 11 日 10 時 23 分 45 秒から 2003 年 10 月 11 日 10 時 23 分 50 秒まで。
-Time 1023451012,1023501013	2003 年 10 月 12 日 10 時 23 分 45 秒から 2003 年 10 月 13 日 10 時 23 分 50 秒まで。
-Time 10234510122002,10235010132002	2002 年 10 月 12 日 10 時 23 分 45 秒から 2002 年 10 月 13 日 10 時 23 分 50 秒まで。

-ProcessID <プロセス ID>

指定されたプロセス ID の情報だけを編集出力します。プロセス ID は 10 進数で指定してください。

-TraceFile <PRF トレースファイル名>

PRF トレースファイルを指定します。トレースファイルは最大 256 個指定できます。このオプションは、-AllTraceFile オプションと同時に指定できません。PRF トレースファイルは、PRF トレースファイル出力先ディレクトリに自動的に作成され、ファイル名は prf_??? になります。??? は、001 からトレースファイルの面数を最大値とした整数です。

PRF トレースファイル出力先ディレクトリ

- Windows の場合
%PRFSPOOL%\%utt%\prf\PRF 識別子 \dcopltrc\
- UNIX の場合
\$PRFSPOOL/utt/prf/PRF 識別子 /dcopltrc/

また、<PRF トレースファイル名> を指定する場合は、絶対パスを指定するか、PRF トレースファイル出力先ディレクトリに移動してから、ファイル名だけを指定します。

-AllTraceFile

カレントディレクトリ下のすべてのファイルを入力ファイルとして読み込みます。このオプションは、-TraceFile オプションと同時に指定できません。

指定できるコマンドオプションの組み合わせの一覧を次に示します。

表 4-2 引数 (コマンドオプション) の組み合わせ一覧 (cprfed コマンド)

コマンド オプション	-h	-Dump	-CSV	-Time	-ProcessID	-TraceFile	-AllTraceFile
-h	-	-h	-h	-h	-h	-h	-h
-Dump	-h	-	-CSV				
-CSV	-h	-CSV	-				
-Time	-h			-			
-ProcessID	-h				-		
-TraceFile	-h					-	x
-AllTraceFile	-h					x	-

(凡例)

- : 同時に指定できます。
- : のあとに続くオプション名のオプションが有効になります。
- x : 同時に指定できません。
- : 該当しません。

戻り値

0 :

PRF デーモンが正常に停止しました。

0 以外 :

コマンド処理中にエラーが発生しました。出力されたメッセージに従って対策したあと、再度コマンドを実行してください。エラーメッセージは標準エラー出力、ログファイルに出力されます。資料の取得方法については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

出力形式

出力形式 1 (ダンプ形式)

```
PRF: aa...aa Process: bb...bb Thread: cc...cc(FF...FF) Trace: dd...dd
ProcessName: zz.zz
Event: eeeee Time: ffff/gg/hh ii:jj:kk lll/mmm/nnn
Rc: oo...oo
ClientAP:pp...pp qq...qq - rr...rr RootAP: ss...ss tt...tt - uu...uu
SendSCD:vv...vv ww...ww RecieveSCD:xx...xx yy...yy
INT: XX...XX OPR: YY...YY
LookupName:ZZ..ZZ
Offset +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f
0123456789abcdef
AAAA BB CC...CC
AAAA BB CC...CC
:
AAAA BB CC...CC
```

4. 性能解析トレースで使用するコマンド

出力形式 2 (CSV 形式)

```
PRF, Process, Thread, Trace, ProcessName, Event, Date, Time, Time (msec/
usec/nsec), Rc, ClientAP IP, ClientAP PID, ClientAP CommNo., RootAP
IP, RootAP PID, RootAP CommNo., SendSCD IP, SendSCD PID, ReceiveSCD
IP, ReceiveSCD PID, INT, OPR, LookupName, OPT, ASCII
aa...aa, bb...bb, cc...cc (CC...CC), dd...dd, zz...zz, ee...ee, ffff/gg/
hh, ii:jj:kk, lll/mmm/nnn, oo...oo, pp...pp, qq...qq, rr...rr, ss...ss, tt...
tt, uu...uu, vv...vv, ww...ww, xx...xx, yy...yy, XX...XX, YY...YY, ZZ...ZZ, DD...
DD, EE...EE
```

(凡例)

記号	説明
aa...aa	レコード状態 (6 文字以内) <ul style="list-style-type: none"> • Rec : レコードの状態は正常です。 • ErrRec : レコードの状態は異常です。
bb...bb	トレース情報を取得したプロセスのプロセス ID (10 進数で 10 けた以内)
cc...cc	トレース情報を取得したプロセス内スレッドのスレッド ID (10 進数で 20 けた以内)
dd...dd	該当スレッドでのトレース通番 (10 進数で進 10 けた以内)
ee...ee	イベント ID (16 進 6 けた (先頭の 0x 含む))
ffff	トレースを取得した時刻 (年)
gg	トレースを取得した時刻 (月)
hh	トレースを取得した時刻 (日)
ii	トレースを取得した時刻 (時)
jj	トレースを取得した時刻 (分)
kk	トレースを取得した時刻 (秒)
lll	トレースを取得した時刻 (ミリ秒)
mmm	トレースを取得した時刻 (マイクロ秒)
nnn	トレースを取得した時刻 (ナノ秒)
oo...oo	リターンコード (16 進数で 10 けた (先頭の 0x 含む))
pp...pp	クライアントアプリケーションの IP アドレス (xxx.xxx.xxx.xxx の形式)
qq...qq	クライアントアプリケーションのプロセス ID (10 進 10 けた以内)
rr...rr	クライアントの通信番号 (18 けた (先頭の 0x 含む))
ss...ss	ルートアプリケーションの IP アドレス (xxx.xxx.xxx.xxx の形式)
tt...tt	ルートアプリケーションのプロセス ID (10 進数で 10 けた以内)
uu...uu	ルートの通信番号 (18 けた (先頭の 0x 含む))
vv...vv	リクエスト要求元 CTM の IP アドレス (xxx.xxx.xxx.xxx の形式)
ww...ww	リクエスト要求元 CTM のプロセス ID (10 進数で 10 けた以内)
xx...xx	リクエスト要求先 CTM の IP アドレス (xxx.xxx.xxx.xxx の形式)
yy...yy	リクエスト要求先 CTM のプロセス ID (10 進数で 10 けた以内)

記号	説明
zz...zz	プロセス名称 (32 文字以内)
XX...XX	インタフェース名 (33 文字以内)
YY...YY	オペレーション名 (33 文字以内)
ZZ...ZZ	ルックアップ名 (33 文字以内) バッチアプリケーションを実行するシステムの場合、スケジュールグループ名を出力します。
AAAA	オフセット
BB	各イベントで取得した情報の 1 バイト分の 16 進数表示。
CC...CC	BB の内容を ASCII 文字で表現します。ASCII 文字で表現できない内容は「.」(ピリオド)で出力します。
DD...DD	各イベントで取得した情報を 512 文字以内の 16 進数形式で出力します。 ダンプ形式の情報は、" で囲んで出力します。
EE...EE	DD の内容を ASCII 文字で出力 (最大 514 文字) します。 ASCII 文字は、" で囲んで出力します。 ASCII 文字で表示できないコード (制御コード) は「.」(ピリオド)で出力し、 ASCII 文字で " (1 文字) は "" (2 文字) で出力します。 CSV 形式の場合に出力される情報です。
FF...FF	スレッドのハッシュ値を出力 (10 進数で 10 けた以内) します。CTM およびリダイレクタで取得したトレース情報には、スレッドのハッシュ値が出力されない場合があります。

注

インタフェース名、オペレーション名、アクセプタ名、ルックアップ名が 33 文字を超える場合は、次のように編集され 33 文字で出力されます。

「前 16 文字」+「*」+「後ろ 16 文字」

「前 32 文字」+「*」

「*」+「後ろ 32 文字」

入力例・出力例

入力例 1

```
cprfed -TraceFile prf_001
```

出力例 1

```
PRF: Rec Process: 516          Thread: 2204          Trace: 10
ProcessName: ctmd.exe
Event: 0x1406 Time: 2004/02/05 16:46:55 576/000/000
Rc: 0
ClientAP: 10.209.13.123    2148          - 0000000000000107 RootAP:
10.209.13.123
2200          - 0000000000000104
SendSCD: 10.209.13.122    516          ReceiveSCD: ****          ****
INT: com/abcd/BookShop          OPR: orderBook
LookupName: BookShop
```

4. 性能解析トレースで使用するコマンド

入力例 2

```
cprfed -Dump -TraceFile prf_001
```

出力例 2

```
PRF: Rec Process: 516          Thread: 2204          Trace: 10
ProcessName: ctmd.exe
Event: 0x1406 Time: 2004/02/05 16:46:55 576/000/000
Rc: 0
ClientAP: 10.209.13.123      2148          - 00000000000000107 RootAP:
10.209.13.123
      2200          - 00000000000000104
SendSCD: 10.209.13.122      516          ReceiveSCD: ****          ****
INT: com/abcd/BookShop          OPR: orderBook
LookupName: BookShop

Offset +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f
0123456789abcdef
+0000 0a d1 0d 7a 04 02 00 00 0a d1 0d 7b 64 08 00 00
...z.....{d...
+0010 00 00 00 00 07 01 00 00 0a d1 0d 7b 98 08 00 00
.....{....
+0020 00 00 00 00 04 01 00 00 00 00 00 00 63 6f 6d 2f
.....com/
+0030 61 6d 61 7a 6f 73 6f 2f 42 6f 6f 6b 53 68 6f 70 abcd/
BookShop
+0040 00 6f 72 64 65 72 42 6f 6f 6b 00 42 6f 6f 6b 53
.orderBook.BookS
+0050 68 6f 70 00 hop.
```

入力例 3

```
cprfed -CSV -TraceFile prf_001
```

出力例 3

```
PRF,Process,Thread,Trace,ProcessName,Event,Date,Time,Time(msec/
usec/nsec),Rc,ClientAP IP,ClientAP PID, ClientAP CommNo.,RootAP
IP,RootAP PID,RootAP CommNo.,SendSCD IP,SendSCD PID,ReceiveSCD
IP,ReceiveSCD PID,INT,OPR,LookupName,OPT,ASCII
Rec,516,2204,10,ctmd.exe,0x1406,2004/02/05,16:46:55,576/000/
000,0,10.209.13.123,2148,0x00000000000000107,10.209.13.123,2200,0
x00000000000000104,10.209.13.122,516,****,****,com/abcd/
BookShop,orderBook,BookShop,
```

注意事項

性能解析トレースでは高負荷時に排他によるボトルネックが生じないように、PRF トレース出力時に排他制御を行っていません。そのため、一部トレース情報が正しく出力されない場合があります。

cprfflush (バッファ内容の強制ファイル出力)

形式

```
cprfflush [-PRFID <PRF識別子>] [ -h ]
```

機能

PRF デモンに対して、バッファに格納されているトレース情報の出力を要求します。

出力する範囲は、前回ファイル出力したトレースの次のトレースから、コマンドが入力された時点までです。

引数

-PRFID <PRF 識別子>

PRF デモンの識別子を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"TSC" や "tsc"、または "CTM" や "ctm" で始まる文字列は指定しないでください。省略した場合は、"PRF_ID" という文字列が PRF 識別子となります。

-h

コマンドの使用方法が表示されます。

戻り値

0 :

正常終了しました。

1 :

PRF デモンが起動していません。PRFSPOOL 環境変数が異なっている、または -PRFID の値が誤っているおそれがあります。

2 :

PRFSPOOL 環境変数が設定されていません。

上記以外 :

コマンド処理中にエラーが発生しました。出力されたメッセージに従って対策したあと、再度コマンドを実行してください。エラーメッセージは標準エラー出力、ログファイルに出力されます。資料の取得方法については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

注意事項

cprfflush コマンドの実行中は、cprfflush コマンドを実行できません。

要求先の PRF デモンと同じ PRFSPOOL 環境変数が設定されている必要があります。設定されていない場合、コマンドがエラーリターンします。

cprfgetpid (PRF デモンプロセス ID の取得)

形式

cprfgetpid [-PRFID <PRF識別子>] [-h]

4. 性能解析トレースで使用するコマンド

機能

指定した PRF 識別子のプロセス ID を取得します。また、PRF デーモンの起動確認を表示します。

引数

-PRFID <PRF 識別子 >

PRF デーモンの識別子を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"TSC" や "tsc"、または "CTM" や "ctm" で始まる文字列は指定しないでください。省略した場合は、"PRF_ID" という文字列が PRF 識別子となります。

-h

コマンドの使用方法が表示されます。

戻り値

0 :

正常終了しました。プロセスは起動していない場合があります。

1 :

プロセスは起動していません。

2 :

PRFSPOOL 環境変数が設定されていません。

上記以外 :

コマンド処理中にエラーが発生しました。出力されたメッセージに従って対策したあと、再度コマンドを実行してください。エラーメッセージは標準エラー出力、ログファイルに出力されます。資料の取得方法については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

入力例・出力例

入力例

```
cprfgetpid
```

出力例

出力結果は、標準出力に出力されます。

```
1700
```

注意事項

PRF デーモンと同じ PRFSPOOL 環境変数が設定されている必要があります。

PRF デーモンが一度も起動していない、または正常終了したあとにこのコマンドを実行した場合は、エラーになります。

cprflevel (PRF トレース取得レベルの表示と変更)

形式

```
cprflevel [-PRFID <PRF識別子>] [ -h ]
          [-PrfChangeLevel <PRFトレース取得レベル> [, [<PRFトレース
          取得レベル>]...]]
          [-PrfLevelIndex <インデックス番号>] [-PrfLevelAll]
```

機能

PRF トレース取得レベルを表示または変更します。

-PrfChangeLevel オプションを省略した場合は、指定した PRF 識別子に指定されている PRF トレース取得レベルを表示します。-PrfChangeLevel オプションを指定した場合は、指定した PRF 識別子に指定されている PRF トレース取得レベルを指定値に変更します。

-PrfLevelIndex オプションを指定した場合は、指定したインデックスの PRF トレース取得レベルの一覧が表示されます。-PrfLevelAll オプションを指定した場合は、すべてのインデックスに対応する PRF トレース取得レベルが表示されます。

引数

-PRFID <PRF 識別子>

PRF デモンの識別子を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"TSC" や "tsc"、または "CTM" や "ctm" で始まる文字列は指定しないでください。省略した場合は、"PRF_ID" という文字列が PRF 識別子となります。

-h

コマンドの使用方法が表示されます。

-PrfChangeLevel <PRF トレース取得レベル> [, [<PRF トレース取得レベル>]...]

PRF トレース取得レベルを 4 バイトの 16 進数 (8 けたの値) で指定します。各インデックス番号に対応するレベル値をコンマ (,) 区切りで左から複数指定できます。該当するインデックスのレベル値を変更しない場合は、値を指定しないでください。また、あるインデックス番号以降の値をすべて指定しない場合は、コンマ (,) の記述を省略してください。インデックス番号 1 のトレース取得レベルを (1)、インデックス番号 2 の PRF トレース取得レベルを (2) とした場合、PRF トレース取得レベルは 「 (1) , (2) , ... 」 と指定します。

PRF トレース取得レベルの指定例を次に示します。

4. 性能解析トレースで使用するコマンド

表 4-3 PRF トレース取得レベルの指定例 (cprflevel コマンド)

PRF トレース取得レベルの指定例	オプションの指定
インデックス番号 1 の PRF トレース取得レベルを指定する	-PrfChangeLevel 0x44445555
インデックス番号 1 とインデックス番号 2 の PRF トレース取得レベルを指定する	-PrfChangeLevel 0x44445555,0x55554444
インデックス番号 2 の PRF トレース取得レベルを指定する	-PrfChangeLevel ,0x55554444

各機能レイヤでの PRF トレース取得レベルの割り当てについて説明します。PRF トレース取得レベルは、インデックスごとに 8 けたの 16 進数で指定します。それぞれのけたには、機能レイヤが二つずつ割り当てられています。16 進数を 2 進数にした場合の上位 2 ビットに割り当てられている機能レイヤを上位レイヤ、下位 2 ビットに割り当てられている機能レイヤを下位レイヤといいます。けたごとの上位レイヤ、下位レイヤに割り当てられている機能レイヤについて、次の表に示します。

表 4-4 詳細レベルのトレース情報を取得する場合の指定値 (cprflevel コマンド)

インデックス	けた番号	上位レイヤ	下位レイヤ
インデックス 1	1 けた目	リダイレクタ	Web コンテナ
	2 けた目	EJB コンテナ	JNDI
	3 けた目	JTA	JCA コンテナ
	4 けた目	DB Connector	RMI
	5 けた目	SOAP 通信基盤	uCosminexus TP1 Connector , TP1/Client/J
	6 けた目	TP1/Message Queue - Access	Cosminexus RM
	7 けた目	OTS	CTM
	8 けた目	CTM	CTM
インデックス 2	1 けた目	HCSC サーバ	標準出力 / 標準エラー出力 / ユーザログ
	2 けた目	DI	uCosminexus Service Coordinator Interactive Workflow
	3 けた目	uCosminexus Service Adapter for Object Access	uCosminexus Service Adapter for Flat Files
	4 けた目	uCosminexus Service Adapter for Message Queue	CCCJPA
	5 けた目	JAX-WS エンジン	CCCJPA コンテナ

インデックス	けた番号	上位レイヤ	下位レイヤ
	6 けた目	CosminexusJMS プロバイダ機能	TP1 インバウンドアダプタ

PRF トレース取得レベルを指定する場合は、けたごとの上位レイヤ、下位レイヤの PRF トレース取得レベルを「標準」「詳細」「保守」のどのレベルで取得するかを決め、それを 16 進数で指定します。上位レイヤと下位レイヤの PRF トレース取得レベルの組み合わせと指定値の対応を、次の表に示します。

なお、保守レベルは、障害発生時などの保守情報を取得するためのレベルです。通常は指定しないでください。

表 4-5 詳細レベルまたは保守レベルのトレース情報を取得する場合の指定値 (cprflevel コマンド)

上位レイヤ	下位レイヤ	指定値
標準	標準	0
標準	詳細	1
標準	保守	2
詳細	標準	4
詳細	詳細	5
詳細	保守	6
保守	標準	8
保守	詳細	9
保守	保守	a

指定例を示します。

表 4-6 PRF トレース取得レベルの指定例 (cprflevel コマンド)

インデックス	指定例	説明
インデックス 1	00000000	インデックス 1 の各機能レイヤの標準レベルのトレース情報を取得します。
	00000004	CTM だけ詳細レベルのトレース情報を取得して、ほかの機能レイヤは標準レベルのトレース情報を取得します。
	55555544	インデックス 1 の各機能レイヤの詳細レベルのトレース情報を取得します。
	a5565544	リダイレクタ、Web コンテナおよび RMI の保守情報を取得して、ほかの機能レイヤは詳細レベルのトレース情報を取得します。
インデックス 2	00000000	インデックス 2 の各機能レイヤの標準レベルのトレース情報を取得します。
	40000000	HCSC サーバだけ詳細レベルのトレース情報を取得します。

4. 性能解析トレースで使用するコマンド

また、RMI については、抑止レベルを指定できます。

RMI に抑止レベルを指定するには、トレース取得レベルの指定値（インデックス 1 の 4 けた目）に次の値を設定してください。

表 4-7 RMI の抑止レベルの指定値

上位レイヤ (DB Connector)	下位レイヤ (RMI)	指定値
標準	抑止	3
詳細	抑止	7
保守	抑止	b

-PrfLevelIndex < インデックス番号 >

変更または参照する PRF トレース取得レベルのインデックス番号を 1 ~ 16 の整数で指定します。デフォルト値は 1 です。この引数と一緒に -PrfChangeLevel を指定した場合は、指定したインデックス番号の PRF トレース取得レベルのレベル値を -PrfChangeLevel で指定した値に変更します。

この引数と一緒に -PrfChangeLevel を指定しなかった場合は、指定したインデックス番号の PRF トレース取得レベルのレベル値が表示されます。この引数と一緒に -PrfChangeLevel を指定する場合は、-PrfChangeLevel には必ず一つの値だけを指定してください。

(例) インデックス番号 2 番目の値を、0x55554444 に変更する場合

```
cprflevel -PrfChangeLevel 0x55554444 -PrfLevelIndex 2
```

-PrfChangeLevel に複数の値を指定した場合は、オプションエラーとなります。

(例) -PrfChangeLevel に複数の値を指定した場合

```
cprflevel -PrfChangeLevel ,0x55554444,0x55554444 -PrfLevelIndex 2
```

-PrfLevelIndex で 2 を指定しているにもかかわらず、-PrfChangeLevel で複数の値を指定しているためです。なお、-PrfLevelIndex の指定を省略した場合は、1 が設定されます。

-PrfLevelAll

すべてのレベル値を表示します。-PrfChangeLevel または -PrfLevelIndex と組み合わせた指定はできません。このオプションと -PrfLevelIndex の組み合わせを、次に示します。

表 4-8 -PrfLevelAll と -PrfLevelIndex の組み合わせ一覧

cprlevel 現在のレベル値の表示方法		-PrfLevelIndex	
		指定	未指定
-PrfLevelAll	指定	オプションエラー	すべてのレベル値を表示
	未指定	指定されたインデックス番号のレベル値だけを表示	インデックス 1 のレベル値だけを表示

指定できるコマンドオプションの組み合わせの一覧を次に示します。

表 4-9 引数 (コマンドオプション) の組み合わせ一覧 (cprlevel コマンド)

コマンドオプション	-h	-PRFID	-PrfChangeLevel	-PrfLevelIndex	-PrfLevelAll
-h	-	-h	-h	-h	-h
-PRFID	-h	-			
-PrfChangeLevel	-h		-		×
-PrfLevelIndex	-h			-	×
-PrfLevelAll	-h		×	×	-

(凡例)

- : 同時に指定できます。
- : のあとに続くオプション名のオプションが有効になります。
- ×: 同時に指定できません。
- : 該当しません。

戻り値

0:

正常終了しました。

1:

PRF デモンが起動していません。PRFSPOOL 環境変数が異なっている, または -PRFID の値が誤っているおそれがあります。

2:

PRFSPOOL 環境変数が設定されていません。

上記以外:

コマンド処理中にエラーが発生しました。出力されたメッセージに従って対策したあと, 再度コマンドを実行してください。エラーメッセージは標準エラー出力, ログファイルに出力されます。資料の取得方法については, マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

4. 性能解析トレースで使用するコマンド

入力例・出力例

入力例 1

```
cprflevel
```

出力例 1

```
KFCT73415-I 2661 1: prf trace level is 0x12345678
```

入力例 2

```
cprflevel -PrfLevelAll
```

出力例 2

```
KFCT73418-I 2661 1: all prf trace levels are:  
[ 1]=0x12345678 [ 2]=0x00000000 [ 3]=0x00000000 [ 4]=0x00000000  
[ 5]=0x00000000 [ 6]=0x00000000 [ 7]=0x00000000 [ 8]=0x00000000  
[ 9]=0x00000000 [10]=0x00000000 [11]=0x00000000 [12]=0x00000000  
[13]=0x00000000 [14]=0x00000000 [15]=0x00000000 [16]=0x00000000
```

入力例 3

```
cprflevel -PrfLevelIndex 5
```

出力例 3

```
KFCT73419-I 2661 1: prf trace level of [5] is 0x00000000.
```

入力例 4

```
cprflevel -PrfChangeLevel 000000001
```

出力例 4

```
KFCT73416-I 2661 1: prf trace level was changed from 0x00000000  
to 0x00000001.
```

入力例 5

```
cprflevel -PrfChangeLevel 0x00000055,0x55554444
```

出力例 5

```
KFCT73420-I 2661 1: prf trace level was changed:  
[ 1]=0x00000000 -> [ 1]=0x00000055  
[ 2]=0x00000000 -> [ 2]=0x55554444
```

入力例 6

```
cprflevel -PrfChangeLevel 0x44445555 -PrfLevelIndex 2
```

出力例 6

```
KFCT73420-I 2661 1: prf trace level was changed:  
[ 2]=0x00000000 -> [ 2]=0x44445555
```

注意事項

現在の PRF トレース取得レベルを表示する場合、`-PrfLevelIndex` と `-PrfLevelAll` の組み合わせで、出力されるメッセージ ID は、「KFCT73415-I」、「KFCT73418-I」、「KFCT73419-I」のどれか一つとなっています。PRF トレース取得レベルを表示する場合の出力されるメッセージ ID の対応は、次のとおりです。

PRF トレース取得レベルを表示する場合のメッセージ ID		-PrfLevelAll	
		指定	未指定
-PrfLevelIndex	指定	-	KAFCT73419-I
	未指定	KAFCT73418-I	KAFCT73415-I

PRF トレース取得レベルを変更する場合、-PrfChangeLevel の指定方法で出力されるメッセージ ID は、「KFCT73416-I」または「KFCT73420-I」のどちらかになります。次の条件をすべて満たした場合、「KFCT73416-I」が出力されます。それ以外は、「KFCT73420-I」が出力されます。

- -PrfChangeLevel 0x44445555 のように値を一つだけ指定し、コンマ(,)を使用していない。
- -PrfLevelIndex を指定していない。

このコマンドは、PRF デーモンの起動中に実行してください。PRF デーモンが起動していないときにコマンドを実行した場合、エラーメッセージを出力します。

変更する PRF デーモンと同じ PRFSPOOL 環境変数が設定されている必要があります。

このコマンドでレベル変更した場合、次のリクエストから変更が有効になります。

cprfstart (PRF デーモンの開始)

形式

```
cprfstart [-h] [-PRFID <PRF識別子>]
           [-PrfTraceLevel <PRFトレース取得レベル> [, [<PRFトレース取得レベル>]...]]
           [-PrfTraceCount <PRFトレースファイル数>]
           [-PrfTraceFileSize <PRFトレースファイルサイズ>]
           [-PrfTraceBufferSize <PRFトレースバッファサイズ>]
           [-PrfRemakeBuffer]
```

機能

PRF デーモン (cprfd) を起動します。なお、PRF デーモンは、ほかのプロセスよりも先に起動してください。

PRF デーモンは開始後にカレントディレクトリを移動します。移動先は、次のディレクトリです。

- Windows の場合
PRF トレース出力ディレクトリ %PRFSPOOL%\%utt%\prf\<PRF 識別子>\
- UNIX の場合
PRF トレース出力ディレクトリ \$PRFSPOOL/utt/prf/<PRF 識別子>/

4. 性能解析トレースで使用するコマンド

引数

-h

コマンドの使用方法が表示されます。

-PRFID <PRF 識別子 >

PRF デモンの識別子を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"TSC" や "tsc", または "CTM" や "ctm" で始まる文字列は指定しないでください。省略した場合は, "PRF_ID" という文字列が PRF 識別子となります。

-PrfTraceLevel <PRF トレース取得レベル > [, [<PRF トレース取得レベル >]...]

-PrfTraceLevel を省略した場合は標準レベルで情報が出力されます。

PRF トレース取得レベルを 4 バイトの 16 進数 (8 けたの値) で指定します。ここで 4 バイトの 16 進数の先頭に 0x を付けることができますが, 0x は無視されます。

各インデックス番号のレベル値をコンマ (,) 区切りで左から複数指定できます。レベル値の指定を省略する場合は, 値を記述しないでください。また, あるインデックス番号以降の値をすべて指定しない場合は, コンマ (,) の記述を省略してください。インデックス番号 1 のトレース取得レベルを (1), インデックス番号 2 の PRF トレース取得レベルを (2) とした場合, PRF トレース取得レベルは 「(1), (2) ...」 と指定します。

PRF トレース取得レベルの指定例を次に示します。

表 4-10 PRF トレース取得レベルの指定例 (cprfstart コマンド)

PRF トレース取得レベルの指定例	オプションの指定
インデックス番号 1 の PRF トレース取得レベルを指定する	-PrfTraceLevel 0x44445555
インデックス番号 1 とインデックス番号 2 の PRF トレース取得レベルを指定する	-PrfTraceLevel 0x44445555,0x55554444
インデックス番号 2 の PRF トレース取得レベルを指定する	-PrfTraceLevel ,0x55554444

各機能レイヤでの PRF トレース取得レベルの割り当てについて説明します。

PRF トレース取得レベルは, インデックスごとに 8 けたの 16 進数で指定します。それぞれのけたには, 機能レイヤが二つずつ割り当てられています。16 進数を 2 進数にした場合の上位 2 ビットに割り当てられている機能レイヤを上位レイヤ, 下位 2 ビットに割り当てられている機能レイヤを下位レイヤといいます。

けたごとの上位レイヤ, 下位レイヤに割り当てられている機能レイヤについて, 次の表に示します。

表 4-11 詳細レベルのトレース情報を取得する場合の指定値 (cprfstart コマンド)

インデックス	けた番号	上位レイヤ	下位レイヤ
インデックス 1	1 けた目	リダイレクタ	Web コンテナ
	2 けた目	EJB コンテナ	JNDI
	3 けた目	JTA	JCA コンテナ
	4 けた目	DB Connector	RMI
	5 けた目	SOAP 通信基盤	uCosminexus TP1 Connector , TP1/Client/J
	6 けた目	TP1/Message Queue - Access	Cosminexus RM
	7 けた目	OTS	CTM
	8 けた目	CTM	CTM
インデックス 2	1 けた目	HCSC サーバ	標準出力 / 標準エラー出力 / ユーザログ
	2 けた目	DI	uCosminexus Service Coordinator Interactive Workflow
	3 けた目	uCosminexus Service Adapter for Object Access	uCosminexus Service Adapter for Flat Files
	4 けた目	uCosminexus Service Adapter for Message Queue	CCCJPA
	5 けた目	JAX-WS エンジン	CCCJPA コンテナ
	6 けた目	CosminexusJMS プロバイダ 機能	TP1 インバウンドアダプタ

PRF トレース取得レベルを指定する場合は、けたごとの上位レイヤ、下位レイヤの PRF トレース取得レベルを「標準」「詳細」「保守」のどのレベルで取得するかを決め、それを 16 進数で指定します。上位レイヤと下位レイヤの PRF トレース取得レベルの組み合わせと指定値の対応を、次の表に示します。

なお、保守レベルは、障害発生時などの保守情報を取得するためのレベルです。通常は指定しないでください。

表 4-12 詳細レベルまたは保守レベルのトレース情報を取得する場合の指定値 (cprfstart コマンド)

上位レイヤ	下位レイヤ	指定値
標準	標準	0
標準	詳細	1
標準	保守	2
詳細	標準	4
詳細	詳細	5

4. 性能解析トレースで使用するコマンド

上位レイヤ	下位レイヤ	指定値
詳細	保守	6
保守	標準	8
保守	詳細	9
保守	保守	a

指定例を示します。

表 4-13 PRF トレース取得レベルの指定例 (cprfstart コマンド)

インデックス	指定例	説明
インデックス 1	00000000	インデックス 1 の各機能レイヤの標準レベルのトレース情報を取得します。
	00000004	CTM だけ詳細レベルのトレース情報を取得して、ほかの機能レイヤは標準レベルのトレース情報を取得します。
	55555544	インデックス 1 の各機能レイヤの詳細レベルのトレース情報を取得します。
	a5565544	リダイレクタ, Web コンテナおよび RMI の保守情報を取得して、ほかの機能レイヤは詳細レベルのトレース情報を取得します。
インデックス 2	00000000	インデックス 2 の各機能レイヤの標準レベルのトレース情報を取得します。
	40000000	HCSC サーバだけ詳細レベルのトレース情報を取得します。

また, RMI については, 抑止レベルを指定できます。

RMI に抑止レベルを指定するには, トレース取得レベルの指定値 (インデックス 1 の 4 けた目) に次の値を設定してください。

表 4-14 RMI の抑止レベルの指定値

上位レイヤ (DB Connector)	下位レイヤ (RMI)	指定値
標準	抑止	3
詳細	抑止	7
保守	抑止	b

-PrfTraceCount <PRF トレースファイル数 >

PRF トレースファイルの面数を 3 ~ 256 の範囲で指定します。省略した場合は, 4 が設定されます。

正常停止中と前回強制停止後の再起動中に, PRF トレースファイルをバックアップします。PRF トレースファイルサイズの合計値が大きい場合, バックアップ処理で PRF デーモンの起動や停止に時間が掛かることがあります。そのため, Management Server でプロセス監視する場合, デフォルトの監視時間ではタイムア

ウトが発生する場合があります。タイムアウトが発生した場合には、監視時間を長くしてください。

-PrfTraceFileSize <PRF トレースファイルサイズ>

PRF トレースファイルの 1 ファイル当たりの最大容量を 1024 ~ 1048576 (単位: キロバイト) の範囲で指定します。ファイル容量を 1 メガバイトとする場合、1024 と指定します。実際のファイルサイズは、この値より多少の増減があります。増減の中で増分の幅は、最大で「-PrfTraceBufferSize 指定値 - 32 キロバイト」です。省略した場合は、8192 が設定されます。

デフォルト値の 8192 より小さい値を指定する場合は、-PrfTraceBufferSize もデフォルト値より小さい値を指定してください。

正常停止中と前回強制停止後の再起動中に、PRF トレースファイルをバックアップします。PRF トレースファイルサイズの合計値が大きい場合、バックアップ処理で PRF デーモンの起動や停止に時間がかかることがあります。そのため、Management Server でプロセス監視する場合、デフォルトの監視時間ではタイムアウトが発生する場合があります。タイムアウトが発生した場合には、監視時間を長くしてください。

-PrfTraceBufferSize <PRF トレースバッファサイズ>

共用メモリに確保するバッファサイズを 512 ~ 102400 (単位: キロバイト) の範囲で指定します。このオプションには、-PrfTraceFileSize で指定した値よりも大きい値は指定できません。省略した場合は、8192 が設定されます。

バッファ領域が不足すると KFCT26999-W メッセージが出力され、PRF トレースが欠落する場合があります。そのため、メッセージが出力されなくなるまでバッファ領域を拡張してください。

-PrfRemakeBuffer

共用メモリを再作成して起動します。再作成できない場合にはエラーとなります。

戻り値

0 :

PRF デーモンが正常に起動しました。

1 :

PRF デーモンの起動に失敗しました。PRFSPOOL 環境変数が設定されていません。

上記以外 :

コマンド処理中にエラーが発生しました。出力されたメッセージに従って対策したあと、再度コマンドを実行してください。エラーメッセージは標準エラー出力、ログファイルに出力されます。資料の取得方法については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

注意事項

PRFSPOOL 環境変数を設定してください。設定していない場合、コマンドがエラーリターンします。

PRF デーモンを再起動する場合、`-PrfTraceCount` と `-PrfTraceFileSize` に前回と同じ値を指定して再起動してください。指定を省略して再起動した場合、PRF トレースファイル数と PRF トレースファイルサイズについては、前回の起動時の値は引き継がれないので、デフォルトの値になります。

PRF デーモンが異常終了して再起動した場合、終了時のトレース取得レベルが引き継がれます。そのため、トレース取得レベルを変更したあとに異常終了した場合、再起動時に指定したトレース取得レベル（前回起動時と同じ値）と異なる値（変更後の値）が設定されます。次に運用例を示します。

1. PRF デーモン起動：`cprfstart -PrfTraceLevel 0x00000001`
2. トレース取得レベル変更（`0x00000001 0x0000000f`）
3. PRF デーモン異常終了
4. PRF デーモン再起動：`cprfstart -PrfTraceLevel 0x00000001`

この場合、再起動後のトレース取得レベルは、`0x0000000f` になります。

PRF デーモンを再起動する場合、前回使用したバッファを再利用し、トレース取得レベルを引き継ぎます。また、バッファのサイズが変更になった場合は、前回のバッファを削除し、再作成します。ほかのプロセスがバッファにアクセスしている場合、この作成処理に失敗します。

PRF デーモンを再起動した場合、共用メモリダンプ出力してから、バッファを初期化します。また、保守情報を次のディレクトリに作成します。

- Windows の場合
`%PRFSPOOL%\utt\prf\<PRF 識別子>\spool\save\`
- UNIX の場合
`$PRFSPOOL/utt/prf/<PRF 識別子>/spool/save/`

Windows の場合、起動済み、または起動中の PRF デーモンと、大文字・小文字だけが異なる PRF 識別子（例えば "aaa" と "AAA"）を持つ PRF デーモンは起動できません。

PRF デーモンが異常終了した場合、PRF デーモンだけを再起動してください。

PRF デーモンが異常終了してから再起動するまでの間に取得されたトレースは破棄されます。

`-PrfTraceCount` または `-PrfTraceFileSize` に大きな値を指定すると、PRF デーモンが異常終了したあとの PRF デーモンの再起動に時間が掛かる場合があります。

cprfstop (PRF デーモンの停止)

形式

```
cprfstop [ -PRFID <PRF識別子> ] [ -Force ][ -h ]
```

機能

PRF デーモン (cprfd) を停止します。なお、PRF デーモンはほかのプロセスよりもあとに停止してください。

引数

-PRFID <PRF 識別子>

PRF デーモンの識別子を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"TSC" や "tsc"、または "CTM" や "ctm" で始まる文字列は指定しないでください。省略した場合は、"PRF_ID" という文字列が PRF 識別子になります。

-Force

PRF デーモンを強制停止します。

-h

コマンドの使用方法が表示されます。

戻り値

0 :

PRF デーモンが正常に停止しました。

1 :

PRF デーモンが起動していません。PRFSPOOL 環境変数が異なっている、または -PRFID の値が誤っているおそれがあります。

2 :

PRFSPOOL 環境変数が設定されていません。

3 :

タイムアウトが発生しました。

上記以外 :

コマンド処理中にエラーが発生しました。出力されたメッセージに従って対策したあと、再度コマンドを実行してください。エラーメッセージは標準エラー出力、ログファイルに出力されます。資料の取得方法については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

注意事項

コマンドがタイムアウトして、PRF デーモンが正常終了しなかった場合は、-Force オ

4. 性能解析トレースで使用するコマンド

プッシュオンを指定して強制停止してください。

停止する PRF デーモンと同じ PRFSPOOL 環境変数が設定されている必要があります。設定されていない場合、コマンドがエラーリターンします。

cprfstart コマンドの `-PrfTraceCount` または `-PrfTraceFileSize` に大きな値を指定すると、PRF デーモンの正常停止に時間が掛かる場合があります。

5

Cosminexus JMS プロバイダ で使用するコマンド

この章では、Cosminexus JMS プロバイダで使用するコマンドの入力形式、機能などについて説明します。

-
- 5.1 Cosminexus JMS プロバイダで使用するコマンドの概要

 - 5.2 Cosminexus JMS プロバイダで使用するコマンドの一覧

 - 5.3 CJMSP ブローカー管理のコマンドの詳細

 - 5.4 コネクションサービス管理のコマンドの詳細

 - 5.5 物理的送信先管理のコマンドの詳細

 - 5.6 永続化サブスクリバ管理のコマンドの詳細

 - 5.7 物理的送信先の設定プロパティ

 - 5.8 Cosminexus JMS プロバイダで使用するコマンドの注意事項
-

5.1 Cosminexus JMS プロバイダで使用するコマンドの概要

Cosminexus JMS プロバイダには、ブローカー構成を編集するために、CJMSP ブローカーユーティリティとコマンドユーティリティがあります。

(1) CJMSP ブローカーユーティリティ

CJMSP ブローカーインスタンスを起動するためのユーティリティで、`cjmsbroker` コマンドを持っています。

(2) コマンドユーティリティ

CJMSP ブローカー、コネクションサービス、物理的送信先、永続化サブスクリバラーおよびトランザクションを管理するためのユーティリティです。管理する対象ごとにコマンドを持っています。

5.2 Cosminexus JMS プロバイダで使用するコマンドの一覧

ここでは、Cosminexus JMS プロバイダで使用するコマンドの一覧について説明します。

5.2.1 CJMSP ブローカーユーティリティのコマンドの一覧

CJMSP ブローカーユーティリティが持つコマンドの一覧を次の表に示します。

表 5-1 CJMSP ブローカーユーティリティのコマンドの一覧

コマンド名称	概要
cjmsbroker	CJMSP ブローカーインスタンスを起動します。

5.2.2 コマンドユーティリティのコマンドの一覧

コマンドユーティリティが持つコマンドの一覧を管理対象ごとに説明します。なお、CJMSP ブローカーの状態が SHUTDOWN_COMPLETE（停止処理が完了した状態）の場合、これらのコマンドは正常に動作しません。

(1) CJMSP ブローカー管理

表 5-2 CJMSP ブローカー管理で使用するコマンドの一覧

コマンド名称	概要
cjmsicmd list bkr	CJMSP ブローカーの状態を一覧表示します。
cjmsicmd pause bkr	稼働中の CJMSP ブローカーを一時停止します。
cjmsicmd query bkr	CJMSP ブローカーのプロパティ値を一覧表示します。
cjmsicmd resume bkr	CJMSP ブローカーを再開します。
cjmsicmd shutdown bkr	CJMSP ブローカーを停止します。

(2) コネクションサービス管理

表 5-3 コネクションサービス管理で使用するコマンドの一覧

コマンド名称	概要
cjmsicmd list svc	利用可能なコネクションサービスを一覧表示します。

(3) 物理的送信先管理

表 5-4 物理的送信先管理で使用するコマンドの一覧

コマンド名称	概要
cjmsicmd compact dst	物理的送信先を圧縮します。
cjmsicmd create dst	物理的送信先を作成します。
cjmsicmd destroy dst	物理的送信先を破棄します。
cjmsicmd list dst	物理的送信先を一覧表示します。
cjmsicmd pause dst	物理的送信先へのメッセージ配信を一時停止します。
cjmsicmd purge dst	物理的送信先からすべてのメッセージを削除します。
cjmsicmd query dst	物理的送信先のプロパティ値を一覧表示します。
cjmsicmd resume dst	物理的送信先へのメッセージ配信を再開します。

(4) 永続化サブスクリバ管理

表 5-5 永続化サブスクリバ管理で使用するコマンドの一覧

コマンド名称	概要
cjmsicmd destroy dur	永続化サブスクリバを破棄します。
cjmsicmd list dur	永続化サブスクリバを一覧表示します。
cjmsicmd purge dur	永続化サブスクリバのすべてのメッセージを削除します。

(5) コマンドユーティリティの共通オプション

コマンドユーティリティが持っているすべてのコマンドに指定できる共通オプションがあります。コマンドユーティリティの共通オプションの一覧を次の表に示します。

表 5-6 コマンドユーティリティの共通オプションの一覧

オプション名称	概要
-rtm timeoutInterval	タイムアウト間隔 (単位: 秒) の初期値を 0 ~ 2147483647 の整数で指定します。コマンドを実行から再度要求を出すまでに CJMSP ブローカーからの応答を待つ時間の初期値です。リトライ間隔はタイムアウト間隔の初期値の倍数が設定されます。デフォルト値は 10 秒です。また、0 を指定した場合は、タイムアウトは発生しません。
-b <ホスト名称>:<ポート番号>	<p>CJMSP ブローカーに接続されているホスト名 (または IP アドレス)、ポート番号を指定します。デフォルト値は次のとおりです。</p> <ul style="list-style-type: none"> ホスト名称: localhost ポート番号: 7676 <p>リモートホストで CJMSP ブローカーインスタンスを起動しているときに正しい値を設定しないと、エラーメッセージが表示されます。ホスト名称を省略してポート番号だけを指定した場合、ローカルホストの CJMSP ブローカーに接続します。例えば、「:7676」と指定した場合、「localhost:7676」と指定した場合と同じ意味になります。</p>

オプション名称	概要
-f	ユーザの確認をしないで、コマンドを実行します。
-h -help	ヘルプを表示します。

5.2.3 Cosminexus JMS プロバイダで使用するコマンドの格納先

Cosminexus JMS プロバイダで使用するコマンドは、次のディレクトリに格納されています。

Windows の場合

<Cosminexus のインストールディレクトリ>\cc\ejmsp\bin\

UNIX の場合

/opt/Cosminexus/cc/ejmsp/bin/

5.3 CJMSP ブローカー管理のコマンドの詳細

CJMSP ブローカー管理で使用するコマンドの入力形式、機能などを次に示します。

cjmsbroker (CJMSP ブローカーの起動)

形式

```
cjmsbroker -name <インスタンス名称>
           -reset store
           -reset messages
           -reset durables
           -remove instance
           -varhome <ロケーション>
           -vmargs <引数1> [[<引数2>] ...]
           -force
           -h | -help
```

機能

CJMSP ブローカーインスタンスを起動します。

引数

-name <インスタンス名称>

起動する CJMSP ブローカーインスタンスの名称を指定します。同じホストで複数の CJMSP ブローカーインスタンスを起動する場合は、異なるインスタンス名称を指定してください。デフォルト値は `cjmsbroker` です。

インスタンスの命名規則を次に示します。

- Windows 環境の場合は、予約デバイス名は指定できません。

(例)

AUX, CON, NUL, PRN, COM1- COM9, LPT1-LPT9

- 指定できる文字数は、OS で規定されたディレクトリ作成時の最大文字数と同じです。
- 英数字と特殊文字を組み合わせて指定します。ただし、Windows 環境の場合では、次に示す特殊文字は使用できません。

¥ / : * ? " < > |

UNIX 環境の場合では、使用可能な文字はファイルシステムの仕様に依存します。

- インスタンス名称に円マーク (¥) またはスラッシュ (/) を使用した場合、動作保証しません。

-reset store

永続化データストアのコンフィグレーション値をリセットします。永続化データストアのコンフィグレーション値をリセットすると、永続化データストアにあるすべての永続化データが削除されます。

永続化データのうち、永続化メッセージだけを削除する場合は `-reset messages` オプションを、永続化サブスクライバーだけを削除する場合は `-reset durables` オプションを指定してください。

`-reset messages`

永続化データストアにある永続化データのうち、永続化メッセージだけを削除します。

`-reset durables`

永続化データストアにある永続化データのうち、永続化サブスクライバーだけを削除します。

`-remove instance`

CJMSP ブローカーインスタンス下にある永続化データストアやその他のファイル、およびディレクトリを削除します。ただし、`props` ディレクトリとログディレクトリは削除されません。

なお、`-remove instance` オプションを使用する場合、次の点に注意してください。

- 削除対象の CJMSP ブローカーインスタンスは起動しないでください。
- `-name` オプションで CJMSP ブローカーインスタンスの名称を指定した場合、指定したインスタンスが削除されます。また、`-name` オプションを指定しない場合、デフォルトの CJMSP ブローカーインスタンス「`ejmsbroker`」が削除されます。
- `ejmsbroker` コマンドは CJMSP ブローカーインスタンスを起動するコマンドですが、`-remove instance` オプションを指定した場合は、CJMSP ブローカーインスタンスは起動されません。
- `-remove instance` コマンドですでに削除されている場合やインスタンスファイルのリポジトリがすでに削除されている場合は、「`Removing instance data`」というメッセージが表示されます。

`-varhome <ロケーション>`

`var` ディレクトリの格納先を絶対パスで指定します。ただし、ネットワーク上のパスは指定できません。デフォルト値は `<CJMSP_HOME>/var` です。

格納先には書き込み権限と読み込み権限の両方を持つディレクトリを指定してください。読み込み権限しか持たないディレクトリを指定すると、エラーメッセージが表示されます。

また、指定したディレクトリが存在しない場合、または利用停止の場合、新しくディレクトリを作成します。

格納先に指定するパスの命名規則を次に示します。

- Windows 環境の場合は、予約デバイス名は指定できません。指定した場合、エラーメッセージが表示されます。

(例)

AUX, CON, NUL, PRN, COM1- COM9, LPT1- LPT9

- 英数字と特殊文字を組み合わせで指定します。ただし、Windows 環境の場合では、次に示す特殊文字は使用できません。

5. Cosminexus JMS プロバイダで使用するコマンド

¥ / : * ? " < > |

UNIX 環境の場合では、使用可能な文字はファイルシステムの仕様に依存します。

使用できない特殊文字を使用した場合、エラーメッセージが表示されます。

- 格納先に指定するパスは Java の標準ファイルの構文規則に従います。

-vmargs <引数>

JavaVM に渡す引数を指定します。指定できる引数は Java HotSpot VM の引数です。指定できる引数の詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」の JavaVM 起動オプションに関する説明を参照してください。

JavaVM へはコマンドラインからだけ引数を渡せます。CJMSP ブローカーインスタンスのコンフィグレーションファイルには、関連する設定プロパティはありません。

次の場合、引数全体をダブルクォーテーション (") で囲みます。

- 引数に空白が含まれている場合
- 引数にイコール (=) が含まれている場合
- 引数を二つ以上指定する場合 (各引数はスペースで区切る)

(例)

```
cjmsbroker -vmargs "-Xms256m -Xmx1024m"
```

- Xms : 最小ヒープサイズ
- Xmx : 最大ヒープサイズ
- m : 容量単位 (メガバイト)

上記の例では、最小ヒープサイズを 256 メガバイト、最大ヒープサイズを 1 ギガバイトに設定します。CJMSP ブローカーの JavaVM のヒープサイズのデフォルト値は 192 メガバイトです。

-force

ユーザの確認をしないで、動作を実行します。

-h | -help

ヘルプを表示します。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

なお、-remove instance オプションを指定した場合は、戻り値として次の値が返ります。

10 :

削除対象のインスタンスが存在しません。

11 :

削除対象のインスタンスは起動中です。

12 :

インスタンスの削除権限がありません。

13 :

インスタンスの削除中に内部エラーが発生しました。

14 :

インスタンスの削除中に I/O エラー発生がしました。

注意事項

インスタンスを作成するとディレクトリに書き込み権限が付与されますが、この権限は変更しないでください。読み込み権限しか持たないディレクトリをインスタンスとして指定すると、エラーメッセージが表示されます。

jms サービスまたは admin サービスのどちらかが起動された場合に、CJMSP ブローカーは起動されます。

cjmsicmd list bkr (CJMSP ブローカーの一覧表示)

形式

```
cjmsicmd list bkr [-b <ホスト名称>:<ポート番号>]
```

機能

CJMSP ブローカーの状態を一覧表示します。

CJMSP ブローカーの状態には次の 3 種類があります。

- OPERATING : CJMSP ブローカーがメッセージを処理できる状態
- SHUTDOWN_STARTED : CJMSP ブローカーの停止処理が開始された状態
- SHUTDOWN_COMPLETE : CJMSP ブローカーの停止処理が完了した状態

詳細については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 基本・開発編 (コンテナ共通機能)」の CJMSP ブローカーの状態の確認に関する説明を参照してください。

上記の状態のうち、一覧に表示できるのは、OPERATING と SHUTDOWN_STARTED です。SHUTDOWN_COMPLETE は一覧に表示できません。

引数

-b < ホスト名称 > : < ポート番号 >

5. Cosminexus JMS プロバイダで使用するコマンド

ホスト名とポート番号を「:」で区切って指定します。

出力例

cjmsicmd list bkr の出力例を次に示します。

```
Cluster is Highly Available   false
-----
      Address                State
-----
DE205372437F45F:7676      OPERATING
KDAN34466-I Successfully listed broker.
```

出力ヘッダ

Address

CJMSP ブローカーが起動時のホストアドレス（ホストシステムのホスト名または IP アドレス）およびポート番号です。

State

CJMSP ブローカーの現在の状態です。

戻り値

0:

正常終了しました。

1:

異常終了しました。

cjmsicmd pause bkr (CJMSP ブローカーの一時停止)

形式

```
cjmsicmd pause bkr [-b <ホスト名称>:<ポート番号>]
```

機能

稼働中の CJMSP ブローカーを一時停止します。

このコマンドを実行して CJMSP ブローカーを一時停止すると、jms サービスだけが停止され、admin サービスは起動されたままになります。そのため、CJMSP ブローカーとしての状態は OPERATING となります。しかし、jms サービスが停止しているため、メッセージは送受信できません。

引数

-b <ホスト名称>:<ポート番号>
 ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:
 正常終了しました。

1:
 異常終了しました。

注意事項

一時停止中の CJMSP ブローカーに対してこのコマンドを実行すると、「Successfully paused the broker」というメッセージが表示されますが、コマンドは無視されます。

cjmsicmd query bkr (CJMSP ブローカーのプロパティ値の一覧表示)

形式

```
cjmsicmd query bkr [-b <ホスト名称>:<ポート番号>]
```

機能

CJMSP ブローカーのプロパティ値を一覧表示します。

引数

-b <ホスト名称>:<ポート番号>
 ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:
 正常終了しました。

1:
 異常終了しました。

cjmsicmd resume bkr (CJMSP ブローカーの再開)

形式

```
cjmsicmd resume bkr [-b <ホスト名称>:<ポート番号>]
```

5. Cosminexus JMS プロバイダで使用するコマンド

機能

CJMSP ブローカーを再開します。

引数

-b <ホスト名称>:<ポート番号>
ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:
正常終了しました。

1:
異常終了しました。

注意事項

一時停止中ではない CJMSP ブローカーに対してこのコマンドを実行すると、
「Successfully resumed the broker」というメッセージが表示されますが、コマンドは
無視されます。

cjmsicmd shutdown bkr (CJMSP ブローカーの停止)

形式

```
cjmsicmd shutdown bkr [-b <ホスト名称>:<ポート番号>]  
                        [-time <待機時間>]
```

機能

CJMSP ブローカーを停止します。

引数

-b <ホスト名称>:<ポート番号>
ホスト名とポート番号を「:」で区切って指定します。

-time <待機時間>
コマンドを実行してから停止されるまでの待機時間を秒単位で指定します。
範囲は 0 ~ 2147483647 の整数で指定します。

戻り値

0:
正常終了しました。

1:

異常終了しました。

注意事項

コマンドを実行してから停止されるまでの間、起動中の jms サービスは引き続き動作します。ただし、新しい jms サービスは起動できません（admin サービスは起動できます）。

コマンドを実行してから停止されるまでの間に、このコマンドを実行すると、CJMSP プロローカーはすぐに停止されます。なお、このコマンド以外のコマンドを実行した場合は、動作を保証しません。

5.4 コネクションサービス管理のコマンドの詳細

コネクションサービス管理で使用するコマンドの入力形式、機能などを次に示します。

cjmsicmd list svc (コネクションサービスの一覧表示)

形式

```
cjmsicmd list svc [-b <ホスト名称>:<ポート番号>]
```

機能

利用可能なコネクションサービスを一覧表示します。

引数

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

出力例

cjmsicmd list svc の出力例を次に示します。

```
-----  
Service Name      Port Number      Service State  
-----  
admin             3616 (dynamic)   RUNNING  
jms                0 (dynamic)      PAUSED  
  
KDAN34113-I Successfully listed services.
```

出力ヘッダ

Service Name

利用可能なサービス名です。

Port number

サービスに割り当てられたポート番号です。

CJMSP プローカーによって割り当てられた、またはユーザが割り当てたポート番号です。

Service State

現在のサービスの状態です。状態は、RUNNING または PAUSED になります。

戻り値

- 0 :
正常終了しました。
- 1 :
異常終了しました。

5.5 物理的送信先管理のコマンドの詳細

物理的送信先管理で使用するコマンドの入力形式、機能などを次に示します。

cjmsicmd compact dst (物理的送信先の圧縮)

形式

```
cjmsicmd compact dst [-t <送信先タイプ>
                      -n <送信先名称>]
                      [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先のファイルベースの永続化ストアデータを圧縮します。

引数

-t <送信先タイプ>

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

-n <送信先名称>

送信先名を指定します。

-t オプションおよび -n オプションを省略した場合、すべての物理的送信先が圧縮されます。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

物理的送信先は圧縮する前に一時停止してください。

cjmsicmd create dst (物理的送信先の作成)

形式

```
cjmsicmd create dst -t <送信先タイプ>
                  -n <送信先名称>
                  [[-o <プロパティ>=<値>]...]
                  [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先を作成します。

引数

-t <送信先タイプ>

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

-n <送信先名称>

送信先名を指定します。

送信先の命名規則は次のとおりです。

- 英数字で指定してください。
- 最初の文字は、英字、アンダースコア (_), ドル記号 (\$) を指定してください。
- 最初の文字に「mq.」および数字は指定できません。
- 「_JMQAdmin」は、Cosminexus JMS プロバイダによって自動生成される物理的送信先の名称であるため指定できません。
- 空白は指定できません。

-o <プロパティ>=<値>

プロパティ名を指定します。

-o オプションで使用できるプロパティを次に示します。

- consumerFlowLimit
- maxNumMsgs
- maxTotalMsgBytes
- maxNumProducers

無効な値を設定した場合は、エラーメッセージが表示されます。

プロパティの詳細については、「5.7 物理的送信先の設定プロパティ」を参照してください。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

一つの CJMSP ブローカーインスタンスに対して、送信先タイプが異なれば、同じ名称の物理的送信先を作成できます。送信先タイプが同じ場合は、同じ名称の物理的送信先は作成できません。

異なる CJMSP ブローカーインスタンスであれば、同じ名称の物理的送信先を作成できます。

CJMSP ブローカー個別プロパティファイルで

`imq.autocreate.queue(topic).consumerFlowLimit` を設定し、`-o` オプションには `consumerFlowLimit` を指定しないでこのコマンドを実行した場合、作成された物理的送信先には、`imq.autocreate.queue(topic).consumerFlowLimit` で指定した数のメッセージを配信できます。`imq.autocreate.queue(topic).consumerFlowLimit` を設定しない場合、配信できるメッセージ数はキューが 100、トピックが 1000 (`consumerFlowLimit` のデフォルト値) となります。

cjmsicmd destroy dst (物理的送信先の破棄)

形式

```
cjmsicmd destroy dst -t <送信先タイプ>  
                    -n <送信先名称>  
                    [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先を破棄します。

引数

`-t <送信先タイプ>`

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

`-n <送信先名称>`

送信先名を指定します。

`-b <ホスト名称>:<ポート番号>`

ホスト名とポート番号を「:」で区切って指定します。

戻り値

0 :
正常終了しました。

1 :
異常終了しました。

注意事項

デッドメッセージキューなどのシステムによって作成された送信先に対して、この操作はできません。

cjmsicmd list dst (物理的送信先の一覧表示)

形式

```
cjmsicmd list dst [-t <送信先タイプ>]
                  [-tmp]
                  [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先を一覧表示します。

送信先タイプの指定がない場合は、キューとトピック両方のタイプの物理的送信先が一覧表示されます。

引数

-t <送信先タイプ>

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

-tmp

一時的な物理的送信先を一覧表示します。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

出力例

cjmsicmd list dst の出力例を次に示します。

5. Cosminexus JMS プロバイダで使用するコマンド

Name	Type	State	Producers	Consumers Total	Count	Msgs UnAck	Avg Size
JMSDEMOQueue	Queue	RUNNING	0	1	1000	451	210.0
JMSDEMOTopic	Topic	RUNNING	0	0	0	0	0.0
mq.sys.dmqs	Queue	RUNNING	0	0	3999	0	702.0

KDAN34110-I Successfully listed destinations.

出力ヘッダ

Name

システムまたはユーザが作成した送信先名です。mq.sys.dmqs は、システムで作成されたデッドメッセージキューです。

Type

送信先の種類です。種類は、キューとトピックがあります。mq.sys.dmqs の送信先はキューです。

State

送信先の状態です。正常な送信先の状態は、RUNNING または PAUSED です。

Producers

送信先のアクティブプロデューサーの数です。

Consumers Total

送信先のアクティブコンシューマーの数です。

Count

送信先に蓄えられたメッセージ数です。

Msgs UnAck

送信先で応答のなかったメッセージ数です。

Avg Size

メッセージの平均サイズです。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

cjmsicmd pause dst (物理的送信先のメッセージ配信の一時停止)

形式

```
cjmsicmd pause dst [-t <送信先タイプ>
                    -n <送信先名称>]
                    [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先に対して、メッセージ配信を一時停止します。

送信先タイプと送信先名称の指定がない場合は、すべての物理的送信先を一時停止します。

引数

-t <送信先タイプ>

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

-n <送信先名称>

送信先名を指定します。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

一時停止中の物理的送信先に対してこのコマンドを実行した場合、無視されます。

デッドメッセージキューを一時停止する場合は、ほかの物理的送信先からのメッセージを受け付けられるようにするため、最後に停止してください。また、再開する場合は、デッドメッセージキューを最初に再開してください。

cjmsicmd purge dst (物理的送信先のメッセージの削除)

形式

```
cjmsicmd purge dst -t <送信先タイプ>  
                  -n <送信先名称>  
                  [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先からすべてのメッセージを削除します。

引数

-t <送信先タイプ>

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

-n <送信先名称>

送信先名を指定します。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:

正常終了しました。

1:

異常終了しました。

cjmsicmd query dst (物理的送信先のプロパティ値の一覧表示)

形式

```
cjmsicmd query dst -t <送信先タイプ>  
                  -n <送信先名称>  
                  [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先のプロパティ値 (単位: バイト) を一覧表示します。

引数

-t <送信先タイプ>

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

-n <送信先名称>

送信先名を指定します。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

戻り値

0:

正常終了しました。

1:

異常終了しました。

cjmsicmd resume dst (物理的送信先へのメッセージ配信の再開)

形式

```
cjmsicmd resume dst [-t <送信先タイプ>
                    -n <送信先名称>]
                    [-b <ホスト名称>:<ポート番号>]
```

機能

物理的送信先へのメッセージ配信を再開します。

送信先タイプと送信先名称の指定がない場合は、すべての物理的送信先を一時停止します。

引数

-t <送信先タイプ>

送信先タイプを指定します。指定できる送信先タイプは次のとおりです。

- q: キュー
- t: トピック

-n <送信先名称>

送信先名を指定します。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

5. Cosminexus JMS プロバイダで使用するコマンド

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

状態が RUNNING の物理的送信先に対してこのコマンドを実行すると、
「Successfully resumed the destination」というメッセージが表示されますが、コマンドは無視されます。

5.6 永続化サブスクライバー管理のコマンドの詳細

永続化サブスクライバー管理で使用するコマンドの入力形式、機能などを次に示します。

cjmsicmd destroy dur (永続化サブスクライバーの破棄)

形式

```
cjmsicmd destroy dur -n <サブスクライバー名称>
                    -c <クライアントID>
                    [-b <ホスト名称>:<ポート番号>]
```

機能

永続化サブスクライバーを破棄します。

引数

- n <サブスクライバー名称>
永続化サブスクライバー名を指定します。
- c <クライアントID>
クライアントIDを指定します。
- b <ホスト名称>:<ポート番号>
ホスト名とポート番号を「:」で区切って指定します。

戻り値

- 0:
正常終了しました。
- 1:
異常終了しました。

cjmsicmd list dur (永続化サブスクライバーの一覧表示)

形式

```
cjmsicmd list dur -d <トピック名称>
                  [-b <ホスト名称>:<ポート番号>]
```

機能

永続化サブスクライバーを一覧表示します。

5. Cosminexus JMS プロバイダで使用するコマンド

引数

-d <トピック名称>

トピック名を指定します。

-b <ホスト名称>:<ポート番号>

ホスト名とポート番号を「:」で区切って指定します。

出力例

cjmsicmd list dur の出力例を次に示します。

トピック名に「Test_Topic」を指定した場合

```
-----  
Durable Sub. Name   Client ID   Number of Messages   Durable Sub. State  
-----  
DurableSUB1        clientID3   2998                  INACTIVE  
KDAN34323-I Successfully listed durable subscriptions.
```

出力ヘッダ

Durable Sub. Name

永続化サブスクリイバー名です。

Client ID

永続化サブスクリイバー用のクライアント ID 値です。

Number of Messages

永続化サブスクリイバーで使用されていないメッセージ数です。

Durable Sub. State

永続化サブスクリイバーの使用状態です。

戻り値

0:

正常終了しました。

1:

異常終了しました。

cjmsicmd purge dur (永続化サブスクリイバーの削除)

形式

```
cjmsicmd purge dur -n <サブスクリイバー名称>  
                  -c <クライアントID>
```

[-b <ホスト名称>:<ポート番号>]

機能

永続化サブスクリバのすべてのメッセージを削除します。

引数

-n <サブスクリバ名称 >

永続化サブスクリバを指定します。

-c <クライアント ID>

クライアント ID を指定します。

-b <ホスト名称>:<ポート番号 >

ホスト名とポート番号を「:」で区切って指定します。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

5.7 物理的送信先の設定プロパティ

ここでは、コマンドユーティリティの `cjmsicmd create dst` コマンド（物理的送信先の作成）の `-o` オプションに指定する設定プロパティについて説明します。

表 5-7 物理的送信先の設定プロパティの一覧

プロパティ	説明
<code>maxNumMsgs</code>	<p>コンシューマーに配信されないメッセージの最大数を <code>-1</code> ~ <code>2147483647</code> の整数で指定します。省略した場合は、<code>-1</code> が設定されます。<code>-1</code> はメッセージ数が無制限であることを示します。デッドメッセージキューの場合は、省略すると <code>1000</code> が設定されます。</p> <p><code>0</code> を指定した場合は、<code>-1</code> とみなされます。</p>
<code>maxTotalMsgBytes</code>	<p>コンシューマーに配信されないメッセージ用の総メモリ量の最大値を <code>-1</code> ~ <code>2147483647</code> の整数で指定します。また、サフィックスとして <code>b</code> (バイト), <code>k</code> (キロバイト), <code>m</code> (メガバイト) を指定できます。</p> <p>省略した場合は、<code>-1</code> が設定されます。デッドメッセージキューの場合は、省略すると <code>10m</code> が設定されます。</p> <p>サフィックスを省略した場合は、<code>b</code> となります。</p> <p><code>-1</code> を指定した場合、メモリ量は無制限となります。</p>
<code>consumerFlowLimit</code>	<p>一つのバッチでコンシューマーに配信されるメッセージの最大数を <code>-1</code> ~ <code>2147483647</code> の整数で指定します。</p> <p>省略した場合は、送信先タイプによって次の値が設定されます。</p> <ul style="list-style-type: none"> • キュー：100 • トピック：1000 • デッドメッセージキュー：1000 <p><code>-1</code> を指定した場合、メッセージの数は無制限になります。</p>
<code>maxNumProducers</code>	<p>プロデューサーの最大数を <code>-1</code> または <code>1</code> ~ <code>2147483647</code> の整数で指定します。省略した場合は、<code>-1</code> (無制限) が設定されます。</p> <p>プロデューサー数が指定した最大値 (<code>-1</code> を除く) に達した場合、新しいプロデューサーは作成できません。</p> <p>なお、<code>0</code> は指定できません。指定した場合は、エラーメッセージが表示されます。</p>

5.8 Cosminexus JMS プロバイダで使用するコマンドの注意事項

ここでは、Cosminexus JMS プロバイダで使用するコマンドの注意事項について説明します。

CJMSP ブローカーでは大文字と小文字を区別します。ただし、OS が Windows の場合、インスタンス名およびコマンドは大文字と小文字を区別しません。例えば、`cjmsbroker` と `CJMSBROKER` は同じ意味になります。

`-o` オプションは、オーバーライドできません。ただし、プロパティの特徴が異なっている場合は、設定できます。同じプロパティが設定された場合、最後に設定されたプロパティが適用されます。

`-h` オプションまたは `-help` オプションを指定した場合、そのほかのオプションは無視されます。

例：`cjmsicmd create dst -n TestQueue -t t -h`

Cosminexus JMS プロバイダのコマンドでは、同じオプションを二つ以上指定した場合、最後のオプションの値が有効となります。

例：`cjmsicmd create dst -n hello -t q -t t`

上記の場合、`-t` オプションを二つ指定していますが、有効になるのは `-t t` です。

無効なオプションを指定してコマンドを実行した場合、指定したオプションが別のコマンドで有効であれば、エラーにはなりません。ただし、オプションの入力形式が間違っている場合は、エラーとなります。

(例 1)

```
cjmsicmd list bkr -n jms
```

`-n` オプションは物理的送信先を指定するオプションで、`cjmsicmd list bkr` コマンドでは無効です。しかし、オプションの入力形式が正しいので、エラーにならないで、`cjmsicmd list bkr` コマンドは正しく動作します。

(例 2)

```
cjmsicmd list bkr -n
```

`-n` オプションは `cjmsicmd list bkr` コマンドでは無効なオプションである上、入力形式にも誤りがあります（送信先が指定されていない）。この場合は、エラーとなります。

6

CTM で使用するコマンド

この章では、CTM で使用するコマンドの入力形式、機能などについて説明します。

6.1 CTM で使用するコマンドの一覧

6.2 CTM で使用するコマンドの詳細

6.1 CTM で使用するコマンドの一覧

CTM で使用するコマンドは、`ctm` で始まります。CTM で使用するコマンドの一覧を、次の表に示します。

表 6-1 CTM で使用するコマンドの一覧

コマンド名称	分類	概要
<code>ctmchpara</code>	スケジュールキューの同時実行数の変更	CTM で管理するスケジュールキューの同時実行数を動的に変更します。
<code>ctmdminfo</code>	CTM ドメイン情報の表示と削除	CTM ドメインに関する情報を表示または削除します。
<code>ctmdmstart</code>	CTM ドメインマネージャの開始	CTM ドメインマネージャを開始します。
<code>ctmdmstop</code>	CTM ドメインマネージャの終了	CTM ドメインマネージャを終了します。
<code>ctmgetpid</code>	プロセス ID の取得	CTM ドメインマネージャ、CTM デーモンのプロセス ID を取得します。
<code>ctmholdque</code>	スケジュールキューの閉塞	スケジュールキューを閉塞します。
<code>ctmjver</code>	実行形式ファイルおよびライブラリのバージョン情報の出力	CTM が提供する jar ファイルのバージョン情報を出力します。
<code>ctmlogcat</code>	メッセージの編集と出力	CTM で出力したメッセージを編集して標準出力に出力します。
<code>ctmlsque</code>	スケジュールキュー情報の出力	スケジュールキューのスケジューリング情報を標準出力に出力します。
<code>ctmrasget</code>	トラブルシュート情報の取得と出力	CTM が必要とするトラブルシュート情報を取得し、出力します。
<code>ctmregltd</code>	CTM レギュレータの開始	CTM レギュレータを開始します。
<code>ctmridinfo</code>	CTM ドメイン情報の出力	CTM ドメイン情報を標準出力に出力します。
<code>ctmrlesque</code>	スケジュールキューの閉塞解除	スケジュールキューの閉塞状態を解除します。
<code>ctmstart</code>	CTM デーモンの開始	CTM デーモンを開始します。
<code>ctmstop</code>	CTM デーモンの終了	CTM デーモンを終了します。
<code>ctmstsed</code>	稼働統計情報の編集と出力	取得した稼働統計情報を必要な情報・出力形式に編集して出力します。
<code>ctmstsflush</code>	バッファ内容の強制ファイル出力	バッファ内のリクエスト情報をファイルに出力します。
<code>ctmstsstart</code>	稼働統計情報の取得開始	稼働統計情報ファイルへの情報の取得を開始します。

コマンド名称	分類	概要
ctmstsstop	稼働統計情報の取得終了	稼働統計情報ファイルへの情報の取得を終了します。
ctmtsegwd	OTM ゲートウェイの開始	OTM クライアントのリクエストを CTM で処理するためのゲートウェイプロセスを開始します。
ctmver	実行形式ファイルおよびライブラリのバージョン情報の出力	CTM が提供する実行形式ファイルおよびライブラリのバージョン情報を出力します。

6.2 CTM で使用するコマンドの詳細

CTM で使用する各コマンドの入力形式，機能などを次に示します。

コマンドの格納先

CTM で使用するコマンドは，次のディレクトリに格納されています。

- Windows の場合
 <Cosminexus のインストールディレクトリ>\¥CTM¥bin¥
- UNIX の場合
 /opt/Cosminexus/CTM/bin/

ctmchpara (スケジュールキューの同時実行数の変更)

形式

```
ctmchpara [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]  
          [-CTMMyHost <ホスト名称またはIPアドレス>]  
          [-CTMTimeOut <タイムアウト値>]  
          -CTMQueue <スケジュールキュー名称>  
          -CTMChangeCount <同時実行数>
```

機能

スケジュールキュー単位の同時実行数を変更します。同時実行数は，スレッド数を増減させることで変更します。

スレッドを増加する場合と減少する場合のコマンド完了については，次のように処理されます。

- スレッドを増加する場合，スレッドの生成完了を待ち合わせます。
- スレッドを減少する場合，リクエストの完了を待ってスレッドを消滅させますが，コマンドは待ち合わせません。

スレッド数はスケジュールキューの状態が次の場合に変更できます。なお，スケジュールキューの状態は，ctmlsqe コマンドで表示される "Status" を確認してください。

- A：スケジューリング可能状態
- H：スケジューリング閉塞中
- C：スケジューリング可能閉塞

また，同時実行数を変更すると，CTM キューを共有するアプリケーションのスレッド数は平均化されます。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

同時実行数を変更する CTM デーモンの CTM ドメイン名称を 1 ~ 31 文字の英数字, またはアンダースコア (_) で指定します。

"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

同時実行数を変更する CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字, ピリオド (.), またはアンダースコア (_) で指定します。

"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は, IP アドレスを指定する場合だけ使用できます。省略した場合は, IP アドレスで CTM 識別子が設定されます。

例えば, ホストの IP アドレスが "172.17.112.43" のとき, "172.17.112.43" という文字列が CTM 識別子となります。

-CTMMyHost <ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。値は 1 ~ 64 文字の文字列で指定します。

省略した場合は, hostname コマンドで取得されるホスト名称が設定されます。

-CTMTimeOut <タイムアウト値 >

スケジュールキューの同時実行数変更処理の待ち時間 (単位 : 秒) を 0 ~ 2147483647 の整数で指定します。"0" を指定した場合, 無限に待ち続けます。省略した場合は "180" (秒) が設定されます。

-CTMQueue <スケジュールキュー名称 >

同時実行数を変更するスケジュールキューの登録名称を 1 ~ 63 文字の英数字, またはアンダースコア (_) で指定します。

-CTMChangeCount <同時実行数 >

同時実行数の変更値を 1 からキューの共有数 × 127 までの整数で指定します。CTM デーモン起動時に指定した -CTMDispatchParallelCount オプションの値 (CTM デーモン起動時に -CTMDispatchParallelCount オプションを省略した場合は "255") より大きい値は指定できません。例えば, 同時実行数を 30 から 10 に変更したい場合は, 10 を指定してください。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生しました。出力されたメッセージに従って対策したあと, 再度コマンドを実行してください。

6. CTM で使用するコマンド

注意事項

バッチアプリケーションを実行するシステムでは、このコマンドを使用しないでください。使用した場合、システムの動作は不定となります。

スケジュールキューを共有している J2EE サーバ数より同時実行数を小さくする場合、各 J2EE サーバにリクエストを送信するスレッド数は 1 となります。

このコマンドを実行するには、Session Bean のプールのインスタンス数を無制限または同時実行数の最大値 "127" 以上の値に設定してください。

J2EE アプリケーションを停止した場合、または J2EE サーバが異常終了した場合の同時実行数は次のように処理されます。

コマンド実行前の場合：

該当する J2EE アプリケーション分の同時実行数を減少させます。

コマンド実行後の場合：

スケジュールキューの同時実行数の設定値は変更されません。

コマンドで設定した同時実行数および実行の成否は、変更先の CTM ログ、または `ctmlsque` コマンドで表示される "ParallelCount" で確認してください。

ctmdminfo (CTM ドメイン情報の表示と削除)

形式

```
ctmdminfo [-h] [-d]
```

機能

ホスト内の CTM 環境に登録済みの CTM ドメインに関する情報を表示または削除します。

引数

-h

コマンドの使用方法が表示されます。

-d

ホスト内の CTM 環境に登録済みの CTM ドメイン情報を削除します。

"ctmdminfo -d" の実行後に、現在の CTM 環境に登録済みの CTMSPOOL と CTM ドメイン名称の一覧が表示されます。一覧から削除したい組み合わせの番号を選択し、コマンドの指示に従って入力します。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

入力例・出力例

1. -d オプションを指定して ctmdminfo コマンドを実行します。

```
ctmdminfo -d
```

2. CTM ドメイン名称および CTMSPOOL の一覧が表示されます。

- Windows の場合

```
No CTM-Domain  Port  CTMSPOOL
1 hitachi      9496  C:¥Program Files¥HITACHI¥ctm¥spool
2 CTMDOMAIN    20137 C:¥Program
Files¥HITACHI¥Cosminexus¥CTM¥spool
Please input Number which you wish delete. ->
```

- UNIX の場合

```
No CTM-Domain  Port  CTMSPOOL
1 hitachi      9496  /home/hitachi/ctm/spool
2 CTMDOMAIN    20137 /opt/Cosminexus/CTM/spool
Please input Number which you wish delete. ->
```

3. 削除したい環境を表す数字を入力します。

```
Please input Number which you wish delete. -> 2
```

4. 問題がなければ "y" を入力します。

- Windows の場合

```
2 CTMDOMAIN    20137 C:¥Program
Files¥HITACHI¥Cosminexus¥CTM¥spool
We will delete this combination of Domain_name and CTMSPOOL
directory.
OK ? (y/n) -> y
```

- UNIX の場合

```
2 CTMDOMAIN    20137 /opt/Cosminexus/CTM/spool
We will delete this combination of Domain_name and CTMSPOOL
directory.
OK ? (y/n) -> y
```

注意事項

同じホストに TPBroker Object Transaction Monitor の環境が存在する場合、TSC ドメインに関する情報も表示または削除できるため注意してください。

ctmdmstart (CTM ドメインマネージャの開始)

形式

```
ctmdmstart [-h] [-CTMDomain <CTMドメイン名称>]
            [-CTMMaxCTM <CTMデーモン管理最大数>]
            [-CTMPort <CTMドメイン管理ポート番号>]
            [-CTMMyHost <ホスト名称またはIPアドレス>]
            [-CTMSendInterval <CTMデーモン情報の送信間隔>]
            [-CTMSubnetMask <サブネットマスク>]
            [-CTMSendHost <ホスト名称またはIPアドレス>]
            [, <ホスト名称またはIPアドレス>, ...]]
            [-CTMSendHostInterval
            <異なるネットワークセグメントへのCTMデーモン情報の送信間隔>]
            [-CTMAliveCheckCount <CTMドメインマネージャ稼働判定係数>]
            [-CTMForceStart | -CTMAutoForce]
```

機能

CTM ドメインマネージャを開始します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM ドメインマネージャが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMMaxCTM <CTM デーモン管理最大数 >

CTM ドメイン内で管理する CTM デーモンの最大数を 1 ~ 1024 の整数で指定します。省略した場合は "64" が設定されます。

-CTMPort <CTM ドメイン管理ポート番号 >

CTM ドメインマネージャが CTM デーモン情報を交換するために使用するポート番号を 5001 ~ 65535 の整数で指定します。同じ CTM ドメイン名称を持つ CTM ドメインマネージャを複数のホストで開始する場合、-CTMPort オプションに同じ値を指定してください。省略した場合はサービス名ファイルが参照されます。サービス名ファイルに指定がないときは "20137" が設定されます。サービス名ファイルへの記述の形式を次に示します。

```
CTM_<CTMドメイン名称>          ポート番号/tcp
```

例えば、"ctmdmstart -CTMDomain JAPAN" と指定して CTM ドメインマネージャを開始する場合、サービス名ファイルの、次のように指定したエントリが読み込まれます。

CTM_JAPAN

10021/tcp

CTM ドメインマネージャは一つのポート番号で tcp および udp の両方のプロトコルを使用します。そのため、CTM ドメインマネージャが使用するポート番号をほかの udp のポート番号として使用しないでください。

-CTMPort オプションの指定値は、異常終了後に CTM ドメインマネージャを再開始するときは変更しないでください。変更したい場合は、CTM ドメインマネージャを正常終了させてから変更してください。

-CTMMyHost < ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを 1 ~ 64 文字の文字列で指定します。省略した場合は、hostname コマンドで取得されるホスト名称が設定されます。

-CTMMyHost オプションの指定値は、異常終了後に CTM ドメインマネージャを再開始するときは変更しないでください。変更したい場合は、CTM ドメインマネージャを正常終了させてから変更してください。

-CTMSendInterval <CTM デーモン情報の送信間隔 >

CTM ドメインマネージャが CTM デーモン情報をネットワークセグメント内にブロードキャストする間隔を秒単位で指定します。値は 0 ~ 65535 の整数で指定してください。CTM デーモン情報をブロードキャストしない場合は、"0" を指定してください。省略した場合は "60 (秒)" が設定されます。

-CTMSendInterval オプションに大きな値を指定すると、異なるホスト上で開始した CTM ドメインマネージャに CTM デーモン情報をブロードキャストする間隔が長くなり、CTM デーモン間での Enterprise Bean の呼び出しに失敗することがあります。小さな値を指定する場合、異なるホスト上で開始した CTM ドメインマネージャに CTM デーモン情報をブロードキャストする間隔が短くなります。その結果、ネットワークに負荷が掛かることがあります。

-CTMSubnetMask < サブネットマスク >

ネットワークのサブネットマスクを、xxx.xxx.xxx.xxx の形式で指定します。xxx には 0 ~ 255 の整数を指定します。区切りにはピリオドを指定します。

省略した場合は "255.255.255.0" が設定されます。

-CTMSendHost < ホスト名称または IP アドレス > [< ホスト名称または IP アドレス >,...]

CTM デーモン情報の配布先となる、異なるネットワークセグメントにある CTM ドメインマネージャが動作するホストのホスト名称または IP アドレスを指定します。値は英数字、ピリオド (.), ハイフン (-), またはアンダースコア (_) で指定します。このオプションを指定することで、複数のネットワークセグメントで構成した CTM ドメインで、CTM デーモン情報を配布できるようになります。

ホスト名称を指定する場合は、<Windows ディレクトリ

6. CTM で使用するコマンド

>¥system32¥drivers¥etc¥hosts (Windows の場合), または etc/hosts (UNIX の場合) に設定されているホスト名称を指定してください。また、ハイフン (-) で始まるホスト名称または IP アドレスは指定しないでください。指定した場合の動作は保証しません。

複数のホストを指定する場合は、指定値をコンマ (,) で区切ります。次に指定例を示します。

```
-CTMSendHost HOSTA,HOSTB
```

<-CTMSendHostInterval <異なるネットワークセグメントへの CTM デーモン情報の送信間隔>

異なるネットワークセグメントにある CTM ドメインマネージャが動作するホストに CTM デーモン情報を送信する間隔を秒単位で指定します。値は 0 ~ 65535 までの整数で指定します。CTM デーモン情報を送信しない場合は、"0" を指定してください。また、省略した場合は、-CTMSendInterval オプションに指定した値が設定されます。

-CTMSendHostInterval オプションに大きな値を指定すると、異なるホスト上で開始した CTM ドメインマネージャに CTM デーモン情報を送信する間隔が長くなり、CTM デーモン間での Enterprise Bean 呼び出しに失敗することがあります。小さな値を指定する場合、異なるホスト上で開始した CTM ドメインマネージャに CTM デーモン情報を送信する間隔が短くなります。その結果、ネットワークに負荷が掛かることがあります。

-CTMSendHostInterval オプションに値を指定しても、-CTMSendHost オプションを指定しなければ無効になります。また、-CTMSendHost、-CTMSendInterval、および -CTMSendHostInterval オプションの指定値によって CTM デーモン情報の送信の有無は異なります。

コマンドオプションの指定値と CTM デーモン情報の送信の関係を次の表に示します。

-CTMSendHost	-CTMSendHostInterval	-CTMSendInterval	CTM デーモン情報
指定あり	0	0	×
		0 以外	×
	0 以外	0	
		0 以外	
	指定なし	0	×
		0 以外	
指定なし	無効	無効	×

(凡例)

: 送信されます。

× : 送信されません。

注

-CTMSendInterval オプションを指定しない場合のデフォルト値 (60 秒) を含みます。

-CTMAliveCheckCount <CTM ドメインマネージャ稼働判定係数 >

異なるホスト上で開始した CTM ドメインマネージャが稼働しているかどうかを判定する係数を、2 から 255 までの整数で指定します。省略した場合は、"2" が設定されます。稼働判定間隔は、次のどちらかの値になります。

- ブロードキャストによって CTM デーモン情報を送信してきた CTM ドメインマネージャの、-CTMSendInterval オプションで指定した <CTM デーモン情報の送信間隔 > に <CTM ドメインマネージャ稼働判定係数 > を掛けた値。
- ネットワークセグメントを超えて CTM デーモン情報を送信してきた CTM ドメインマネージャの、-CTMSendHostInterval オプションで指定した <異なるネットワークセグメントへの CTM デーモン情報の送信間隔 > に <CTM ドメインマネージャ稼働判定係数 > を掛けた値。

稼働判定間隔の間に CTM デーモン情報が送信されなかった場合、送信元の CTM ドメインマネージャが停止したと判定して、送信元に対する CTM デーモン情報は削除されます。CTM デーモン情報が削除された場合、その CTM デーモンへのリクエストの振り分けはされません。

-CTMForceStart

前回は強制終了または異常終了した場合に、CTM ドメインマネージャを強制的に正常開始します。前回管理していた CTM デーモンを強制終了し、管理情報を初期化して CTM ドメインマネージャを開始します。このとき、CTM デーモンは再開始されません。

前回は正常終了していた場合は正常開始するため、-CTMForceStart オプションの指定は無視されます。

-CTMForceStart オプションは、-CTMAutoForce オプションと同時に指定できません。

-CTMAutoForce

前回は強制終了または異常終了した場合に、前回の管理情報を引き継いで CTM ドメインマネージャを自動強制開始します。前回の管理情報を正常に引き継げる場合は、終了前のシステム状態を復元できます。前回の管理情報を引き継げない場合は、-CTMForceStart オプションが指定されたときと同じように動作します。

前回は正常終了していた場合は正常開始するため、-CTMAutoForce オプションの指定は無視されます。

-CTMAutoForce オプションは、-CTMForceStart オプションと同時に指定できません。

戻り値

0 :

6. CTM で使用するコマンド

正常終了しました。

0 以外：

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

注意事項

CTM ドメインマネージャは開始後にカレントディレクトリを移動します。そのため、次に示す場合のコアファイルの出力先は、実行形式ファイルのディレクトリとは異なります。

- "ctmdmstop -CTMForce -CTMDebugInfo", kill コマンドなどで CTM ドメインマネージャを強制終了した場合
- CTM ドメインマネージャが異常終了した場合

移動後のカレントディレクトリを次に示します。

- Windows の場合
"%CTMSPOOL%\adm\CTM ドメイン名称 %"
- UNIX の場合
"\$CTMSPOOL/adm/CTM ドメイン名称 /"

ctmdmstop (CTM ドメインマネージャの終了)

形式

```
ctmdmstop [-h] [-CTMDomain <CTMドメイン名称>]
           [-CTMMyHost <ホスト名称またはIPアドレス>]
           [-CTMForce [-CTMDebugInfo]]
```

機能

CTM ドメインマネージャを終了します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM ドメインマネージャが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。

"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMMyHost <ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを 1 ~ 64 文字の文字列で指定します。省略した場合は、

hostname コマンドで取得されるホスト名称が設定されます。

-CTMMyHost オプションの指定値は、異常終了後に CTM ドメインマネージャを再開始するときは変更しないでください。変更したい場合は、CTM ドメインマネージャを正常終了させてから変更してください。

-CTMForce

CTM ドメインマネージャを強制終了します。

-CTMForce オプションで CTM ドメインマネージャを強制終了させたあとの再開始では、ctmdmstart コマンドに -CTMForceStart オプションを指定して開始してください。

-CTMDebugInfo

CTM ドメインマネージャの強制終了時にデバッグ情報として、共用メモリダンプおよびコアを取得します。-CTMDebugInfo オプションは、-CTMForce オプションを指定した場合にだけ有効です。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

ctmgetpid (プロセス ID の取得)

形式

```
ctmgetpid [-h] | -CTMDomain <CTMドメイン名称> [-CTMID <CTM識別子>]
          [-CTMMyHost <ホスト名称またはIPアドレス>]
```

機能

プロセス監視を実行する場合に使用します。CTM ドメインマネージャ、CTM デーモンのプロセス ID を取得します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

-CTMID オプションを指定する場合、プロセス ID 取得対象となる CTM デーモンが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。

6. CTM で使用するコマンド

-CTMID オプションを省略する場合、プロセス ID 取得対象となる CTM ドメインマネージャの CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。

指定するときは、"CTM" または "ctm" で始まる CTM ドメイン名称は指定しないでください。

-CTMID <CTM 識別子 >

プロセス ID 取得対象となる CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (.), またはアンダースコア (_) で指定します。

"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、-CTMDomain オプションで指定された CTM ドメインマネージャのプロセス ID が取得されます。

-CTMMyHost <ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。値は 1 ~ 64 文字の文字列で指定します。

省略した場合は、hostname コマンドで取得されるホスト名称が設定されます。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

ctmholdque (スケジュールキューの閉塞)

形式

```
ctmholdque [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
            [-CTMMyHost <ホスト名称またはIPアドレス>]
            {-CTMQueue <スケジュールキュー登録名称> | -CTMQueueALL}
            [[-CTMRequestLeave] [-CTMTimeOut <タイムアウト値>]
             | [-CTMChangeServer]]
            [-CTMForce]
```

機能

スケジュールキューを閉塞します。

スケジュールキューを閉塞すると、それ以降のリクエストとスケジュールキュー内のリクエストはエラーになります。ただし、-CTMRequestLeave オプションを指定した場合、スケジュールキュー内のリクエストは閉塞状態でも J2EE アプリケーションで処理されます。

また、`-CTMChangeServer` オプションを指定して閉塞した場合、オンライン状態で J2EE アプリケーションを入れ替えられます。

引数

`-h`

コマンドの使用方法が表示されます。

`-CTMDomain <CTM ドメイン名称 >`

スケジュールキューを閉塞する CTM デーモンの CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (`_`) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

`-CTMID <CTM 識別子 >`

スケジュールキューを閉塞する CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (`.`)、またはアンダースコア (`_`) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (`.`) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、IP アドレスで CTM 識別子が設定されます。

例えば、ホストの IP アドレスが "172.17.112.43" のとき、"172.17.112.43" という文字列が CTM 識別子となります。

`-CTMMyHost <ホスト名称または IP アドレス >`

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。値は 1 ~ 64 文字の文字列で指定します。省略した場合は、`hostname` コマンドで取得されるホスト名称が設定されます。

`-CTMQueue <スケジュールキュー登録名称 >`

閉塞するスケジュールキューの登録名称を 1 ~ 63 文字の英数字、またはアンダースコア (`_`) で指定します。

バッチアプリケーションを実行するシステムでは、スケジュールキューの登録名称としてスケジュールグループ名を指定してください。

`-CTMQueueALL`

すべてのスケジュールキューを一括して閉塞します。このとき、閉塞済みのスケジュールキューも含まれます。

`-CTMRequestLeave`

スケジュールキュー閉塞時に、スケジュールキューに滞留しているリクエストを破棄しないでスケジュールキューを閉塞します。破棄しなかったリクエストは、J2EE アプリケーションで処理されます。

`ctmholdque` コマンドは滞留しているすべてのリクエストが取り出されるのを、

`-CTMTimeOut` オプションの指定値まで待ち合わせます。リクエストの待ち合わせ

6. CTM で使用するコマンド

中に `ctmrlesque` コマンドによって閉塞解除された場合は、リクエストの受け付けを可能にして、`ctmholdque` コマンドは正常終了します。

-CTMTimeOut <タイムアウト値>

スケジュールキューの閉塞処理の待ち時間（単位：秒）を 0 ~ 2147483647 の整数で指定します。"0" を指定した場合、無限に待ち続けます。省略した場合は "180" (秒) が設定されます。

-CTMChangeServer

スケジュールキューから J2EE アプリケーションへのリクエストのスケジュールを閉塞します。このとき、EJB クライアントからのリクエストをキューイングできます。また、スケジュールキューに滞留しているリクエストは、`ctmrlesque` コマンドによって閉塞解除されるまで保持されます。ただし、CTM デーモンを停止する場合には、スケジュールキューに滞留しているリクエストを破棄して停止処理を続行します。

-CTMChangeServer オプションでスケジュールキューを閉塞した場合、リクエストのキューイングおよび滞留しているリクエストを保持するため、オンライン状態で J2EE アプリケーションを入れ替えられます。なお、同じ CTM ドメイン内にスケジュール可能な J2EE アプリケーションが存在する場合には、負荷が掛かっていない J2EE アプリケーションに対応するスケジュールキューにスケジュールします。

-CTMChangeServer オプションは、-CTMRequestLeave オプションおよび

-CTMTimeOut オプションと同時に指定できません。

J2EE サーバが異常終了したときのリクエスト保持機能 (`ctmstart`

-CTMQueueDeleteWait) と併用した場合、J2EE サーバが異常終了したあとに `ctmholdque` -CTMChangeServer は実行できません。

-CTMForce

スケジュールキュー閉塞時に、CTM デーモンに通信を行わないでスケジュールキューを高速に閉塞します。その際、滞留しているリクエストは一定時間後に破棄されます。ただし、同時に -CTMRequestLeave オプションを指定すると、滞留しているリクエストは破棄されません。また、同時に -CTMChangeServer オプションを指定した場合、-CTMChangeServer オプションは無視されます。

戻り値

0 :

正常終了しました。

1 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

2 :

指定されたスケジュールキューはすでに閉塞しています。なお、この戻り値は

-CTMQueue オプションを指定した場合だけ返却されます。

ctmjver (実行形式ファイルおよびライブラリのバージョン情報の出力)

形式

```
ctmjver -h | <ファイルパス>
```

機能

CTM が提供する jar ファイルのバージョン情報を出力します。

引数

-h

コマンドの使用方法が表示されます。

<ファイルパス>

対象となるファイルパス (相対パスおよび絶対パス) を指定します。

出力形式

```
Information For : aa...aa
Product Name : bb...bb
PP Number : cc...cc
Version : dd...dd
Copyright : ee...ee
Company : ff...ff
Build OS : gg...gg
Build Compiler : hh...hh
Build Date : ii...ii jj...jj
Build Number : kk...kk
```

(凡例)

- aa...aa : ファイル名称 (文字列)
- bb...bb : 製品名称 (文字列)
- cc...cc : PP 型名 (文字列)
- dd...dd : バージョンおよびリビジョン (文字列)
- ee...ee : コピーライト (文字列)
- ff...ff : 会社名 (文字列)
- gg...gg : ビルド OS (文字列)
- hh...hh : ビルドコンパイラ (文字列)
- ii...ii : ビルド日付 (文字列)
- jj...jj : ビルド時刻 (文字列)
- kk...kk : 保守情報 (文字列)

6. CTM で使用するコマンド

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

ctmlogcat (メッセージの編集と出力)

形式

```
ctmlogcat [-h] | [-d | -t <CTM識別子>]
```

機能

CTM で出力したメッセージを編集して標準出力に出力します。オプションによって CTM ログを出力するか、CTM ドメインログを出力するかを指定できます。

引数

-h

コマンドの使用方法が表示されます。

-d

CTM ドメインログを出力します。

-t <CTM 識別子 >

指定した CTM 識別子の CTM ログを出力します。値は 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。省略した場合は、IP アドレスが設定されます。

すべてのオプションを省略した場合は、IP アドレスを CTM 識別子とした -t オプションが設定されます。マルチホームホスト環境で CTM ログを出力する場合は、-t オプションは省略できません。

例えば、ホストの IP アドレスが "192.17.113.19" の場合、"-t 192.17.113.19" が設定されます。

戻り値

0 :

正常終了しました。出力するデータがない (KFOT73591-I メッセージが出力された) 場合、および -h オプションを指定した場合も 0 が返却されます。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

注意事項

ctmlogcat コマンド実行時には、CTMSPOOL 環境変数が設定されている必要があります。

- -t オプションを指定する場合、該当する CTM 識別子の動作している実行環境に設定した CTMSPOOL 環境変数が必要です。
- -d オプションを指定する場合、CTM ドメインマネージャ実行時に指定した CTMSPOOL 環境変数が必要です。

ctmlsque (スケジュールキュー情報の出力)

形式

```
ctmlsque [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
          [-CTMMyHost <ホスト名称またはIPアドレス>]
          [-CTMQueue <スケジュールキュー登録名称>]
          [-CTMInterface [<インタフェース名称>]] [-CSV]
          [-CTMAppInfo]
```

機能

スケジュールキューのスケジューリング情報を出力します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

スケジュールキュー情報を出力する CTM デーモンの CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

スケジュールキュー情報を出力する CTM デーモンの識別子を 1 ~ 31 文字の英数字、ピリオド (.)、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、IP アドレスで CTM 識別子が設定されます。

例えば、ホストの IP アドレスが "172.17.112.43" のとき、"172.17.112.43" という文字列が CTM 識別子となります。

6. CTM で使用するコマンド

-CTMMyHost < ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。値は 1 ~ 64 文字の文字列で指定します。省略した場合は、hostname コマンドで取得されるホスト名称が設定されます。

-CTMQueue < スケジュールキュー登録名称 >

出力するスケジュールキューの登録名称を 1 ~ 63 文字の英数字、またはアンダースコア (_) で指定します。

バッチアプリケーションを実行するシステムでは、スケジュールキューの登録名称としてスケジュールグループ名を指定してください。

-CTMInterface < インタフェース名称 >

出力する EJB のインタフェース名称を 1 文字以上の文字列で指定します。インタフェース名称を指定した場合、指定したインタフェース名称と EJB の INS 登録名称が出力されます。

-CTMInterface オプションだけを指定した場合、すべてのインタフェース名称と INS 登録名称が出力されます。

-CSV

標準出力情報を CSV 形式で出力します。

-CTMAppInfo

スケジュールキューを共有しているアプリケーション情報を出力します。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

出力形式

標準出力形式

```
*** ctmd Information ***
```

```
CTMDomain          aa...aa
CTMID               bb...bb
```

```
-----
----
```

```
*** CTMQueue Information ***
```

```
QueueName          cc...cc
Status              d
ApplicationCount    ee...ee
RequestCount        ff...ff
RequestMax          gg...gg
QueueLength         hh...hh
```

```

ParallelCount          ii...ii
TotalThreadCount      jj...jj
*** Application Information ***
[1] J2EEServerName     kk...kk
[1] StartParallelCount ll...ll
[1] ResidentThreadCount mm...mm

[2] J2EEServerName     kk...kk
[2] StartParallelCount ll...ll
[2] ResidentThreadCount mm...mm
*** LookupName and Interface Information ***
[1] LookupName         nn...nn
[1] InterfaceName      oo...oo

[2] LookupName         nn...nn
[2] InterfaceName      oo...oo

[3] LookupName         nn...nn
[3] InterfaceName      oo...oo
-----
----
*** CTMQueue Information ***
QueueName             cc...cc
Status                d
ApplicationCount      ee...ee
RequestCount          ff...ff
RequestMax            gg...gg
QueueLength           hh...hh
ParallelCount         ii...ii
TotalThreadCount      jj...jj
*** LookupName and Interface Information ***
:
:

```

CSV 出力形式

```

CTMDomain,CTMID
aa...aa,bb...bb
QueueName,Status,ApplicationCount,RequestCount,RequestMax,QueueLength,ParallelCount,TotalThreadCount,J2EEServerName,StartParallelCount,ResidentThreadCount,LookupName,InterfaceName
cc...cc,dd...dd,ee...ee,ff...ff,gg...gg,hh...hh,ii...ii,jj...jj,kk...kk,ll...ll,mm...mm,nn...nn,oo...oo
,,,,,,,,,kk...kk,ll...ll,mm...mm,nn...nn,oo...oo
,,,,,,,,,nn...nn,oo...oo
:
:
QueueName,Status,ApplicationCount,RequestCount,RequestMax,QueueLength,ParallelCount,TotalThreadCount,J2EEServerName,StartParallelCount,ResidentThreadCount,LookupName,InterfaceName
cc...cc,dd...dd,ee...ee,ff...ff,gg...gg,hh...hh,ii...ii,jj...jj,kk...kk,ll...ll,mm...mm,nn...nn,oo...oo
,,,,,,,,,kk...kk,ll...ll,mm...mm,nn...nn,oo...oo
,,,,,,,,,nn...nn,oo...oo
:
:

```

(凡例)

6. CTM で使用するコマンド

記号	説明	種別	
aa...aa	指定した CTM ドメイン名称 (1 ~ 31 文字の英数字)	CTM デーモン情報	
bb...bb	指定した CTM 識別子 (1 ~ 31 文字の英数字)		
cc...cc	指定した条件と一致するスケジュールキュー名称 (1 ~ 63 文字の英数字)	スケジュールキュー情報 (1)	
d	指定した条件と一致するスケジュールキューの状態を次の値で示します。 <ul style="list-style-type: none"> • S: スケジュールキュー準備中 • A: スケジューリング可能状態 • E: スケジュールキュー終了処理中 • H: スケジュールキュー閉塞中 • C: キューイング可能閉塞 		
ee...ee	スケジュールキューを共有している J2EE アプリケーション数 (10 進数 5 けた)		
ff...ff	スケジュールキューに滞留しているリクエスト数 (10 進数 10 けた)		
gg...gg	スケジュールキューに滞留したリクエスト最大数 (10 進数 10 けた)		
hh...hh	同時に登録できるリクエスト数 (10 進数 5 けた)		
ii...ii	スケジュールキューが管理している同時実行数 (10 進数 3 けた)		
jj...jj	スケジュールキューが管理している現在の総常駐スレッド数 (10 進数 3 けた)		
kk...kk	スケジュールキューが接続している J2EE サーバ名称 (英数字)		スケジュールキュー情報 (2)
ll...ll	kk...kk で表示された J2EE アプリケーションの開始時の同時実行数 (10 進数 3 けた)		
mm...mm	kk...kk で表示された J2EE アプリケーションに対する現在の常駐スレッド数 (10 進数 3 けた)		
nn...nn	キューが管理している INS 登録名称 (英数字)	スケジュールキュー情報 (3)	
oo...oo	キューが管理しているインタフェース名称 (英数字)		

注

- 指定した条件と一致するスケジュールキューが複数ある場合、一致するスケジュールキューの数だけ、スケジュールキュー情報 (1) ~ (3) が繰り返し出力されます。
- -CTMInterface オプションを指定した場合、スケジュールキュー情報 (1) の情報出力後、スケジュールキューが管理している INS 登録名、インタフェース名情報をすべて出力するまで、スケジュールキュー情報 (3) の情報が繰り返し出力されます。
- -CTMAppInfo オプションを指定した場合、スケジュールキュー情報 (1) の情報出力後、スケジュールキューが接続している J2EE サーバ名称、J2EE アプリケーション開始時のスレッド数、J2EE アプリケーションのスレッド数をすべて出力するまで、スケジュールキュー情報 (2) の情報が繰り返し出力されます。
- -CTMInterface オプションと -CTMAppInfo オプションの両方を指定した場合、スケジュールキュー情報 (1) の情報出力後、スケジュールキュー情報 (2) とスケジュールキュー情報 (3)

の情報が繰り返し出力されます。

- バッチアプリケーションを実行するシステムでは、「出力形式」の(凡例)の記載は一部異なります。バッチアプリケーションを実行するシステムでは、J2EE アプリケーションはバッチアプリケーションとなり、J2EE サーバはバッチサーバとなります。また、スケジュールキュー名称および INS 登録名称には、スケジュールグループ名が出力されます。

ctmrasget (トラブルシューティング情報の取得と出力)

形式

```
ctmrasget { [-h] | [-p] [-s] [-CTMSPool <CTMスプールディレクトリ> |
            -CTMDomain <CTMドメイン名称>]
            [-PRFSPool <PRFデーモン実行環境ディレクトリ>] <出力先ディレクトリ>}
```

機能

ctmrasget コマンドを実行するホスト上で CTM が必要とするトラブルシューティング情報をファイルに取得し、取得した情報の一覧を標準出力または標準エラー出力に出力します。

引数

-h

コマンドの使用方法が表示されます。

-p

Cosminexus TPBroker についてのトラブルシューティング情報を出力先ディレクトリに出力します。あらかじめ、Cosminexus TPBroker の VBROKER_ADM 環境変数を設定してください。

-s

共通メッセージ (eventlog または syslog) の取得を抑制します。

-CTMSPool <CTM スプールディレクトリ>

トラブルシューティング情報を取得する CTM スプールディレクトリを指定します。

-CTMDomain <CTM ドメイン名称>

トラブルシューティング情報を取得する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。

省略した場合は、ホスト内の CTM 環境に登録済みのすべての CTM ドメインからトラブルシューティング情報を取得します。

-PRFSPool <PRF デーモン実行環境ディレクトリ>

トラブルシューティング情報を取得する PRF デーモン実行環境ディレクトリを指定します。省略した場合は PRFSPOOL 環境変数の値が設定されます。このオプションを省略し、かつ PRFSPOOL 環境変数が設定されていない場合は、PRF デーモンの情報は取得されません。

6. CTM で使用するコマンド

<出力先ディレクトリ>

トラブルシュート情報を出力するファイルのディレクトリ名（相対パスまたは絶対パス）を指定します。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

出力形式

```
Cosminexus Component Transaction Monitor: RAS completed collection of aa...aa
```

(凡例)

aa...aa : 収集する情報、ファイル名称など

注意事項

出力先ディレクトリにすでに同名のファイルがある場合は上書きされます。そのため、トラブルシュート情報を残しておきたい場合は、ファイルをコピーしておくか、新しく作成したディレクトリを出力先ディレクトリに指定してください。

ctmregltd (CTM レギュレータの開始)

形式

```
ctmregltd [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
          [-PRFID <PRF識別子>] [-CTMMyHost <ホスト名称またはIPアドレス>]
          [-CTMClientConnectCount <コネクション数>]
          [-CTMEjbPort <EJBリクエスト受信ポート番号>]
          [-CTMAgent {1|0}]
          [-CTMTimeOut <タイムアウト値>]
```

機能

EJB クライアントからの要求を分散集約するために、CTM レギュレータを開始します。CTM レギュレータは CTM デーモンが動作するホストと同じホストで開始してください。なお、CTM レギュレータを直接開始することはできません。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM デーモンが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (.), またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、IP アドレスが CTM 識別子に設定されます。例えば、ホストの IP アドレスが "172.17.112.43" のとき、"172.17.112.43" という文字列が CTM 識別子となります。

-PRFID <PRF 識別子 >

PRF トレースを出力する PRF デーモンの識別子を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"TSC" や "tsc" または "CTM" や "ctm" で始まる文字列は指定しないでください。省略した場合は、"PRF_ID" という文字列が PRF 識別子となります。

-CTMMyHost < ホスト名称または IP アドレス >

マルチホームホストまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを 1 ~ 64 文字の文字列で指定します。省略した場合は、hostname コマンドで取得されるホスト名称が設定されます。

-CTMID オプションを省略して -CTMMyHost オプションを指定した場合、デフォルトの CTM 識別子は -CTMMyHost オプションで指定した IP アドレスになります。

-CTMClientConnectCount < コネクション数 >

CTM レギュレータに接続できるコネクション数を 1 ~ 32767 の整数で指定します。コネクション数は、EJB クライアントからの接続数です。EJB クライアントが create()、invoke() を繰り返す場合は、EJB クライアントのプロセス終了、通信障害、またはタイムアウトが発生するまで CTM レギュレータとのコネクションが確立されます。したがって、-CTMClientConnectCount には、EJB クライアントのプロセス数分の設定が必要になる場合があります。省略した場合は、ctmstart コマンドに設定した -CTMClientConnectCount が引き継がれます。ctmstart コマンドで -CTMClientConnectCount を省略した場合は "64" が設定されます。

-CTMEjbPort <EJB リクエスト受信ポート番号 >

EJB クライアントが CTM レギュレータの新しいコネクションを探すときに使用するポート番号を 5001 ~ 65535 の整数で指定します。省略した場合は、OS が自動的に割り当てたポート番号を使用します。

-CTMAgent {1|0}

EJB に対応する CORBA オブジェクトを TPBroker の OSAgent に登録するかどうかを指定します。ORB ゲートウェイ機能を使用する場合は、"1" を指定します。省

6. CTM で使用するコマンド

略した場合、または "0" が指定された場合は、OSAgent に登録しません。

-CTMTimeOut <タイムアウト値>

クライアントからのサービス要求を受信してからサービスの応答が返るまでのデフォルト待ち時間（単位：秒）を 0 ~ 2147483647 の整数で指定します。"0" を指定した場合、無限に待ち続けます。省略した場合は "180"（秒）が設定されます。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

注意事項

CTM レギュレータは開始後にカレントディレクトリを移動します。そのため、次に示す場合のコアファイルの出力先は、実行形式ファイルのディレクトリとは異なります。

- "ctmstop -CTMForce -CTMDebugInfo", kill コマンドなどで CTM レギュレータを強制終了した場合
- CTM レギュレータが異常終了した場合

移動後のカレントディレクトリを次に示します。

- Windows の場合
"%CTMSPPOOL%\%adm%\CTM 識別子 \reg\ カレントディレクトリ ID%"
- UNIX の場合
"\$CTMSPPOOL/adm/CTM 識別子 /reg/ カレントディレクトリ ID/"

カレントディレクトリ ID は、KFCT70606-I メッセージを参照してください。

ctmridinfo (CTM ドメイン情報の出力)

形式

```
ctmridinfo [-h] | [-CTMDomain <CTMドメイン名称>]
            [-CTMHost <ホストの完全修飾名またはIPアドレス>]
            [-CTMMyHost <ホスト名称またはIPアドレス>]
```

機能

CTM ドメイン情報を出力します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM ドメイン情報を出力する CTM ドメインマネージャの CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMHost <ホストの完全修飾名または IP アドレス >

-CTMDomain オプションで指定した CTM ドメイン名称に対応する CTM ドメインマネージャがあるホストの完全修飾名または IP アドレスを指定します。値は 1 ~ 64 文字の文字列で指定します。マルチホームホスト環境では、CTM の使用する

-CTMMyHost オプションに指定するホスト名称または IP アドレスを指定してください。

省略した場合は、自ホストの IP アドレスが設定されます。また、マルチホームホスト環境では、省略時に設定される IP アドレスは OS の動作に依存します。

-CTMMyHost <ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。値は、1 ~ 64 文字の文字列で指定します。省略した場合は、hostname コマンドで取得されるホスト名称が設定されます。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

ctmrlsque (スケジュールキューの閉塞解除)

形式

```
ctmrlsque [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
          [-CTMMyHost <ホスト名称またはIPアドレス>]
          {-CTMQueue <スケジュールキュー登録名称> | -CTMQueueALL}
```

機能

スケジュールキューの閉塞状態を解除します。

引数

-h

コマンドの使用方法が表示されます。

6. CTM で使用するコマンド

-CTMDomain <CTM ドメイン名称 >

スケジュールキューを閉塞解除する CTM デーモンの CTM ドメイン名称を 1 ~ 31 文字の英数字, またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

スケジュールキューを閉塞解除する CTM デーモンの識別子を 1 ~ 31 文字の英数字, ピリオド (.), またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は, IP アドレスを指定する場合だけ使用できます。省略した場合は, IP アドレスが CTM 識別子に設定されます。例えば, ホストの IP アドレスが "172.17.112.43" のとき, "172.17.112.43" という文字列が CTM 識別子となります。

-CTMMyHost <ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。値は 1 ~ 64 文字の文字列で指定します。省略した場合は, hostname コマンドで取得されるホスト名称が設定されます。

-CTMQueue <スケジュールキュー登録名称 >

閉塞解除するスケジュールキューの登録名称を 1 ~ 63 文字の英数字, またはアンダースコア (_) で指定します。

バッチアプリケーションを実行するシステムでは, スケジュールキューの登録名称としてスケジュールグループ名を指定してください。

-CTMQueueALL

すべてのスケジュールキューを一括して閉塞解除します。閉塞解除済みのスケジュールキューを含めることができます。

戻り値

0 :

正常終了しました。

1 :

コマンド処理中にエラーが発生したため, 異常終了しました。出力されるメッセージに従って対策したあと, 再度コマンドを実行してください。

2 :

指定されたスケジュールキューはすでに閉塞解除しています。なお, この戻り値は, -CTMQueue オプションを指定した場合だけ返却されます。

ctmstart (CTM デーモンの開始)

形式

```
ctmstart [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
          [-CTMINSRef <CORBAネーミングサービスのURL>]
          [-CTMDCSendTimeOut<CTMデーモン間リクエスト転送時のタイムアウト値
>]
          [-PRFID <PRF識別子>]
          [-CTMPort <ポート番号>]
          [-CTMEjbPort <EJBリクエスト受信ポート番号>]
          [-CTMMyHost <ホスト名称またはIPアドレス>]
          [-CTMRegStart <プロセス数>]
          [-CTMRegOption <CTMコマンドオプションファイル名称>]
          [-CTMEnviron <CTMユーザ環境変数定義ファイル名称>]
          [-CTMStartTimeOut <タイムアウト値>]
          [-CTMEntryCount <登録プロセス数>] [-CTMNice <nice値>]
          [-CTMClientConnectCount <接続クライアントの最大数>]
          [-CTMServerConnectCount <接続J2EEサーバの最大数>]
          [-CTMLogFileSize <CTMログファイル最大サイズ>]
          [-CTMLogFileCount <CTMログファイル最大数>]
          [-CTMQueueCount <スケジュールキュー登録数>]
          [-CTMQueueRegistCount <スケジュールキュー共有数>]
          [-CTMServerCacheSize <サーバキャッシュサイズ>]
          [-CTMDispatchPolicy normalDispatch | priorDispatch]
          [-CTMDispatchParallelCount <スレッドの最大値>]
          [-CTMMaxRequestCount <最大リクエスト数>]
          [-CTMWatchRequest<タイムアウト発生回数, 監視時間間隔>]
          [-CTMQueueDeleteWait <スケジュールキュー保留時間>]
          [-CTMCreatePolicy normalDispatch | priorDispatch]
          [-CTMLoadCheckInterval <負荷状況監視時間間隔>]
          [-CTMWatchQueue
<監視開始滞留率, システム停止処理率, 監視時間間隔, 停止モード>]
          [-CTMTSCGwStart <プロセス数>]
          [-CTMTSCGwOption <CTMコマンドオプションファイル名称>]
          [-CTMStatsUse {Y|N}]
          [-CTMStatsFileCount <稼働統計情報ファイルの世代数>]
          [-CTMStatsFileSize <稼働統計情報ファイルサイズ>]
```

機能

リクエストをスケジューリングするための CTM デーモンを開始します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM デーモンが属する CTM ドメイン名称を 1 ~ 31 文字の英数字, またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM ドメイン名称は指定しないでください。また, CTM 識別子と CTM ドメイン名称には同じ値を指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

6. CTM で使用するコマンド

-CTMID <CTM 識別子 >

CTM デーモンの識別子を 1 ~ 31 文字の英数字、ピリオド (.), またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。また、同じ CTM ドメイン内で重複する CTM 識別子は使用できません。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、IP アドレスが CTM 識別子に設定されます。

例えば、ホストの IP アドレスが "172.17.112.43" のときは、"172.17.112.43" という文字列が CTM 識別子となります。

-CTMINSRef <CORBA ネーミングサービスの URL >

CTM デーモンと対応付ける CORBA ネーミングサービスの URL を指定します。値は次の形式で 1 ~ 65535 文字で指定します。

<ホスト名称>:<ポート番号>

次に指定例を示します。

```
-CTMINSRef localhost:9001
```

このオプションは省略できません。

CORBA ネーミングサービスは、CTM デーモンと同じホスト上に開始してください。また、J2EE サーバやそのほかの CTM デーモンが使用している CORBA ネーミングサービスを指定しないでください。それらの CORBA ネーミングサービスを共用した場合の動作は保証しません。

-CTMDCSendTimeOut<CTM デーモン間リクエスト転送時のタイムアウト値 >

ほかの CTM デーモンへのリクエスト転送処理の待ち時間を、0 ~ 2147483647 の整数 (単位: 秒) で指定します。"0" を指定した場合、無限に待ち続けます。省略した場合は "180" (秒) が設定されます。このオプションのタイムアウト値では、ほかの CTM デーモンからのリクエストの応答を監視しません。ホストやネットワークに障害が発生した場合、デーモン間転送処理が行なわれると、このオプションで指定したタイムアウト値が経過するまで、障害を検知できません。タイムアウト値を短くすると、長大電文のデーモン間転送に失敗することがあります。

-PRFID <PRF 識別子 >

PRF トレースを出力する PRF デーモンの識別子を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"TSC" や "tsc" または "CTM" や "ctm" で始まる文字列は指定しないでください。省略した場合は、"PRF_ID" という文字列が PRF 識別子となります。

-CTMPort <ポート番号 >

CTM デーモンの新しいコネクションを探すときに使用するポート番号を指定します。値は 5001 ~ 65535 の整数で指定します。省略した場合はサービス名ファイル

が参照されます。サービス名ファイルに指定がないときは "20138" が設定されます。サービス名ファイルへの記述の形式を次に示します。

```
CTM_CTMDメイン名称CTM識別子      ポート番号/tcp
```

例えば, "ctmstart -CTMDomain JAPAN -CTMID Hitachi" と指定して CTM デーモンを開始する場合, サービス名ファイルの次のように指定したエントリが読み込まれます。

```
CTM_JAPANHitachi                  10021/tcp
```

-CTMEjbPort <EJB リクエスト受信ポート番号 >

EJB クライアントが CTM デーモンの新しいコネクションを探すときに使用するポート番号を指定します。値は 5001 ~ 65535 の整数で指定します。省略した場合は, OS が自動的に割り当てたポート番号を使用します。

-CTMMyHost <ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを指定します。値は 1 ~ 64 文字の文字列で指定します。省略した場合は, hostname コマンドで取得されるホスト名称が設定されます。

-CTMID オプションを省略して -CTMMyHost オプションを指定した場合, デフォルトの CTM 識別子は -CTMMyHost オプションで指定した IP アドレスになります。

-CTMRegStart <プロセス数 >

CTM デーモン開始時に自動開始する CTM レギュレータのプロセス数を 0 ~ 32767 の整数で指定します。CTM デーモンは CTM レギュレータの動作を監視し, 異常終了を検出すると再開始します。省略した場合は, "1" が設定されます。

J2EE サーバを開始する場合は, CTM レギュレータを必ず開始してください。自動開始の CTM レギュレータの数が多くなると, CTM デーモンの開始処理に時間が掛かる場合があります。その場合は, -CTMStartTimeOut オプションで開始処理の待ち時間を調整してください。

-CTMRegOption <CTM コマンドオプションファイル名称 >

CTM レギュレータを自動開始するときに指定するコマンドオプションを記述したファイル名称を指定します。

- Windows の場合
英数字, ピリオド (.), 円符号 (¥), コロン (:), プラス (+), ハイフン (-), またはアンダースコア (_) で指定します。
- UNIX の場合
英数字, ピリオド (.), スラッシュ (/), プラス (+), ハイフン (-), またはアンダースコア (_) で指定します。

CTM コマンドオプションファイル名称はユーザ任意の名称です。コマンドオプションファイルには英数字, 空白, ハイフン (-), ピリオド (.), スラッシュ (/), また

6. CTM で使用するコマンド

はアンダースコア (_) を記述できます。コマンドオプション以外は記述しないでください。例えば、次のように指定します。

```
-CTMClientConnectCount 64
```

CTM コマンドオプションファイル名称を "regoption" とする場合は、CTM デーモン開始時に次のように指定します。

```
ctmstart -CTMRegOption regoption -CTMRegStart 10
```

自動開始する複数の CTM レギュレータのポート番号を固定したい場合は、コマンドオプションファイル内で -CTMEjbPort オプションを指定します。指定された -CTMEjbPort オプションの値を基底値とし、1 ずつ加算した値が各 CTM レギュレータの -CTMEjbPort オプションに指定されます。

コマンドオプションファイルに指定しなかった -CTMEjbPort、および -CTMPort 以外のオプションは、ctmstart コマンドに指定された値を引き継いで設定します。

-CTMEnviron <CTM ユーザ環境変数定義ファイル名称 >

開始するプロセスが使用する環境変数を記述した CTM ユーザ環境変数定義ファイル名称を絶対パスで指定します。

- Windows の場合

英数字、ピリオド (.), 円符号 (¥), コロン (:), プラス (+), ハイフン (-), またはアンダースコア (_) で指定します。

- UNIX の場合

英数字、ピリオド (.), スラッシュ (/), プラス (+), ハイフン (-), またはアンダースコア (_) で指定します。

CTM ユーザ環境変数定義ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」を参照してください。

-CTMStartTimeOut <タイムアウト値 >

CTM デーモンの開始処理の待ち時間 (単位 : 秒) を 0 ~ 2147483647 の整数で指定します。"0" を指定した場合、無限に待ち続けます。省略した場合は "180" (秒) が設定されます。

-CTMEntryCount <登録プロセス数 >

CTM デーモンで管理するプロセスの数を 32 ~ 32767 の整数で指定します。

省略した場合、-CTMServerConnectCount オプションと

-CTMClientConnectCount オプションの指定値の合計が設定されます。

-CTMNice <nice 値 >

CTM デーモンが管理するサーバで使用する -CTMNice オプションの指定値のデフォルトを整数で指定します。省略した場合は "0" が設定されます。指定値の範囲

は OS が提供する nice の指定範囲に従います。

なお、-CTMNice オプションは UNIX にしかありません。

- CTMClientConnectCount <CTM レギュレータと OTM ゲートウェイの最大プロセス数 >
CTM デーモンに接続する CTM レギュレータと OTM ゲートウェイの最大プロセス数を 0 ~ 32767 の整数で指定します。省略した場合は "64" が設定されます。
- CTMServerConnectCount < 接続 J2EE サーバの最大数 >
CTM デーモンに接続する J2EE サーバの最大数を 0 ~ 32767 の整数で指定します。省略した場合は "64" が設定されます。
- CTMLogFileSize <CTM ログファイル最大サイズ >
CTM ログファイルの 1 ファイルの最大サイズをメガバイト単位で指定します。値は 1 ~ 3 までの整数で指定します。省略した場合は "1" (メガバイト) が設定されます。
(-CTMLogFileSize オプションの指定値) × (-CTMLogFileCount オプションの指定値) メガバイト分のメッセージを CTM ログファイルに保存できます。CTM ログファイルに保存するメッセージの容量を変更するには、まず -CTMLogFileCount オプションの指定値を変更してください。-CTMLogFileCount オプションの指定値の変更だけでは不十分な場合に、-CTMLogFileSize オプションを指定してください。
- CTMLogFileCount <CTM ログファイル最大数 >
CTM ログファイルの最大ファイル数を 2 ~ 32 の整数で指定します。省略した場合は "2" が設定されます。
CTM ログファイルのサイズが -CTMLogFileSize オプションの指定値を超えると、CTM ログファイルは次のファイルに切り替えられます。また、CTM ログファイルの切り替え時に CTM ログファイルの総数が -CTMLogFileCount オプションの指定値を超えていると、最も古い CTM ログファイルの内容が削除されます。これらの内容に留意して CTM ログファイルの容量を見積もってください。
- CTMQueueCount <スケジュールキュー登録数 >
CTM デーモンに登録できるスケジュールキュー数を 1 ~ 32767 の整数で指定します。省略した場合は "128" が設定されます。
- CTMQueueRegistCount <スケジュールキュー共有数 >
同じスケジュールキューを共有できる J2EE アプリケーション数を 1 ~ 32767 の整数で指定します。省略した場合は "64" が設定されます。
- CTMServerCacheSize <サーバキャッシュサイズ >
CTM ドメイン内のサーバ情報を格納するキャッシュテーブルのサイズ (単位: キロバイト) を 1 ~ 32767 の整数で指定します。省略した場合は "1024" (キロバイト) が設定されます。
サーバキャッシュテーブルのサイズは次に示す計算式で求めてください。

6. CTM で使用するコマンド

```
{ 256 × 同じCTMドメイン内のCTMデーモン数
+ 256 × 同じCTMドメイン内の全スケジュールキュー数
+ (256 + INS登録名の最大長) × 同じCTMドメイン内のスケジュールキューに登録するEnterprise Bean数
+ (256 + インタフェース名の最大長) × 同じCTMドメイン内のインタフェース数
} × 1.5 (バイト)
```

-CTMDispatchPolicy normalDispatch | priorDispatch

スケジュールポリシーを次の値で指定します。

- "normalDispatch"

スケジュールキューの負荷状況に応じてリクエストをスケジュールします。

- "priorDispatch"

リクエストを受け付けた CTM デーモンに登録されているスケジュールキューに、優先的にリクエストをスケジュールします。ただし、その CTM デーモンに登録されているスケジュールキューが高負荷状態、および閉塞状態の場合は、ほかの CTM デーモンに登録しているスケジュールキューにリクエストをスケジュールします。省略した場合は "priorDispatch" が設定されます。

-CTMDispatchParallelCount < スレッドの最大値 >

CTM デーモン内でクライアントからの要求メッセージを J2EE サーバに振り分けるスレッドの最大値、つまり、CTM デーモンに登録される J2EE アプリケーションの平行カウント（常駐するスレッド数）の和を指定します。値は 0 ~ 32767 の整数で指定します。省略した場合は "255" が設定されます。

-CTMMaxRequestCount < 最大リクエスト数 >

CTM デーモンに登録するスケジュールキューが、同時に登録できるリクエスト数（キューの長さ）を 1 ~ 32767 の整数で指定します。省略した場合は "50" が設定されます。

-CTMWatchRequest < タイムアウト発生回数、監視時間間隔 >

CTM デーモンがタイムアウト時間監視を実施します。指定された閉塞時間間隔内で、指定されたタイムアウト発生回数以上のタイムアウトが発生した場合に、スケジュールキューを閉塞します。閉塞処理は時間監視のタイミングで実施されるため、指定されたタイムアウトが発生後、次の監視時に閉塞します。この機能によってリクエストキューが閉塞された場合、キューに滞留しているリクエストを破棄し、負荷情報の更新を行います。

タイムアウト発生回数

自動閉塞するためのタイムアウト発生回数を、1 ~ 32767 の整数（単位：回）で指定します。

監視時間間隔

タイムアウト発生回数を求める時間間隔を、0 ~ 32767 の整数（単位：秒）で指定します。0 を指定した場合は、タイムアウト閉塞機能は動作しません。

-CTMQueueDeleteWait < スケジュールキュー保留時間 >

異常終了した J2EE サーバが登録していたスケジュールキューを削除しないで保留し、J2EE アプリケーションの再開始を待つ場合の待ち時間（単位：秒）を指定します。値は 0 ~ 2147483647 の整数で指定します。

指定した時間内に J2EE アプリケーションが再開始された場合は、保留中のスケジュールキューを再利用し、キューイングされていたリクエストがスケジュールされます。

指定時間を経過しても J2EE アプリケーションが再開始されない場合は、スケジュールキューを閉塞したあと、スケジュールキューに登録されたリクエストを削除します。なお、スケジュールキューの閉塞、およびスケジュールキューに登録されたリクエストの削除は、インターバル処理で実行されるため、指定時間よりも遅れることがあります。

省略した場合、または "0" が指定された場合は、スケジュールキューを保留しないでスケジュールキューに登録されたリクエストを削除します。

オンライン状態での J2EE アプリケーションの入れ替え機能（ctmholdque

-CTMChangeServer）と併用した場合、ctmholdque -CTMChangeServer が実行されたあとに J2EE サーバが異常終了すると、この機能は無効となります。

-CTMCreatePolicy normalDispatch | priorDispatch

create 要求の CTM デーモン選択ポリシーを次の値で指定します。

省略した場合は "normalDispatch" が設定されます。

- "normalDispatch"

スケジュールキューの負荷状況に応じて CTM デーモンを選択します。

- "priorDispatch"

create 要求を受け付けた CTM デーモンを、優先的に選択します。ただし、その CTM デーモンに登録されているスケジュールキューが高負荷状態、および閉塞状態の場合は、ほかの CTM デーモンを選択します。

-CTMLoadCheckInterval < 負荷状況監視時間間隔 >

スケジュールキューの負荷状況を監視する時間間隔を、0 ~ 32767 の整数（単位：秒）で指定します。

指定した時間間隔ですべてのスケジュールキューの負荷状況を確認します。負荷状況に変化がある場合は、変化のあるスケジュールキュー単位に最新の負荷状況を同一ドメイン内のすべての CTM デーモンに通知します。

省略した場合、10 が設定されます。0 が指定された場合、一定時間間隔での負荷状況監視を行いません。

-CTMWatchQueue < 監視開始滞留率、システム停止処理率、監視時間間隔、停止モード >

スケジュールキューの滞留監視の条件を設定します。

-CTMWatchQueue オプションを省略した場合、スケジュールキューの滞留監視は行いません。

-CTMWatchQueue オプションで指定する引数は、必ず次の順番でコンマ(,)で区

6. CTM で使用するコマンド

切って指定します。引数を省略することはできません。

監視開始滞留率

キュー滞留監視状態へ移行する滞留率のしきい値を、0 ~ 99 の整数（単位：%）で指定します。0 を指定した場合は、常に監視します。

システム停止処理率

CTM デーモンの停止へ移行する処理率のしきい値を、1 ~ 100 の整数（単位：%）で指定します。

監視時間間隔

CTM デーモン停止しきい値の処理率を求める時間間隔を、0 ~ 32767 の整数（単位：秒）で指定します。0 を指定した場合は、スケジュールキューの滞留監視は行いません。

停止モード

スケジュールキューの滞留を検知した場合に、CTM デーモンを停止するかどうかを指定します。

- 1 :
CTM デーモンを停止します。
- 0 :
CTM デーモンを停止しません。

-CTMTSCGwStart < プロセス数 >

CTM デーモン開始時に自動開始する OTM ゲートウェイのプロセス数を 0 ~ 32767 の整数で指定します。省略した場合は、"0" が設定されます。

CTM デーモンは OTM ゲートウェイの動作を監視し、異常終了を検出すると再開します。

自動開始の OTM ゲートウェイの数が多くなると、CTM デーモンの開始処理に時間が掛かる場合があります。その場合は、-CTMStartTimeOut オプションで開始処理の待ち時間を調整してください。

-CTMTSCGwOption < CTM コマンドオプションファイル名称 >

OTM ゲートウェイを自動開始するときに指定するコマンドオプションを記述したファイル名称を絶対パスで指定します。

- Windows の場合
英数字、ピリオド (.), 円符号 (¥), コロン (:), プラス (+), ハイフン (-), またはアンダースコア (_) で指定します。
- UNIX の場合
英数字、ピリオド (.), スラッシュ (/), プラス (+), ハイフン (-), またはアンダースコア (_) で指定します。

CTM コマンドオプションファイル名称はユーザ任意の名称です。コマンドオプションファイルには、英数字、空白、ハイフン (-), ピリオド (.), スラッシュ (/), またはアンダースコア (_) を記述できます。コマンドオプション以外は記述しないで

ください。例えば、次のように指定します。

```
-CTMClientConnectCount 64
```

CTM コマンドオプションファイル名称を "tscoption" とする場合は、CTM デーモン開始時に次のように指定します。

```
ctmstart -CTMTSCGwOption tscoption -CTMTSCGwStart 1
```

自動開始する複数の OTM ゲートウェイのポート番号を固定したい場合は、コマンドオプションファイル内で -CTMPort オプションを指定します。指定された -CTMPort オプションの値を基底値とし、1 ずつ加算した値が各 OTM ゲートウェイの -CTMPort オプションに指定されます。

コマンドオプションファイルに指定しなかった -CTMPort 以外のオプションは、ctmstart コマンドに指定された値を引き継いで設定します。

-CTMStatsUse {Y|N}

該当する CTM ノードで稼働統計情報を取得するかどうかを指定します。小文字での指定もできます。省略した場合は "Y" が設定され、該当する CTM ノードで稼働統計情報を取得します。

-CTMStatsFileCount <稼働統計情報ファイル世代数>

稼働統計情報ファイルの世代数を 3 ~ 256 の整数で指定します。省略した場合は "3" が設定されます。

-CTMStatsFileSize <稼働統計情報ファイルサイズ>

稼働統計情報ファイルのサイズ (単位: メガバイト) を 1 ~ 1024 の整数で指定します。省略した場合は "3 (メガバイト)" が設定されます。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

注意事項

CTM デーモンは、開始後にカレントディレクトリを移動します。そのため、次に示す場合のコアファイルの出力先は、実行形式ファイルのディレクトリとは異なります。

- "ctmstop -CTMForce -CTMDebugInfo", kill コマンドなどで CTM デーモンを強制終了した場合
- CTM デーモンが異常終了した場合

6. CTM で使用するコマンド

移動後のカレントディレクトリを次に示します。

- Windows の場合
"%CTMSPOOL%\¥adm¥CTM 識別子 ¥"
- UNIX の場合
"\$CTMSPOOL/adm/CTM 識別子 /"

-CTMRegStart オプションを指定した場合、CTM レギュレータの自動開始が一つでも異常終了すると、CTM デーモンは開始処理失敗と判断してエラーメッセージを出力し、開始処理を中止します。

-CTMRegOption オプションで指定したコマンドオプションファイル内に記述したコマンドオプションの正当性を、自動開始した CTM レギュレータがチェックします。そのため、コマンドオプション不正、および -CTMEjbPort オプション指定によるポート番号の重複は、自動開始した CTM レギュレータで出力したエラーメッセージを参照して対策してください。

CTM デーモンの開始完了前のメッセージには、-CTMLogFileSize オプションおよび -CTMLogFileCount オプションが有効にならない場合があります。

ctmstop (CTM デーモンの終了)

形式

```
ctmstop [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
         [-CTMMyHost <ホスト名称またはIPアドレス>]
         [-CTMForce [-CTMDebugInfo]]
```

機能

リクエストをスケジューリングするための CTM デーモンを終了します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM デーモンが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (.)、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、IP アドレスが CTM 識別子に設定されます。例えば、ホストの IP

アドレスが "172.17.112.43" のとき, "172.17.112.43" という文字列が CTM 識別子となります。

-CTMMyHost <ホスト名称または IP アドレス>

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを 1 ~ 64 文字の文字列で指定します。省略した場合は, hostname コマンドで取得されるホスト名称が設定されます。

-CTMID オプションを省略して -CTMMyHost オプションを指定した場合, デフォルトの CTM 識別子は -CTMMyHost オプションで指定した IP アドレスになります。

-CTMForce

CTM デーモンを強制終了します。

-CTMDebugInfo

CTM デーモンの強制終了時にデバッグ情報として, 共用メモリダンプおよびコアを取得します。

-CTMDebugInfo オプションは, -CTMForce オプションを指定したときだけ有効です。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため, 異常終了しました。出力されるメッセージに従って対策したあと, 再度コマンドを実行してください。

ctmstsed (稼働統計情報の編集と出力)

形式

```
ctmstsed [-h] [-CTMedit <編集出力形式>[,<編集出力形式>...]]
          [-CTMInterval <時間間隔>] [-CTMTime [<開始時刻>][,<終了時刻>]]
          [-CTMQueueName <キュー名>[,<キュー名>...]]
          [-CTMMethod <メソッド名>[,<メソッド名>...]]
          [-CTMCsv] [-CTMStdout]
          [<稼働統計情報ファイル名>[ <稼働統計情報ファイル名>...]]
```

機能

稼働統計情報ファイルに, 取得した稼働統計情報を編集して出力します。デフォルトはテキスト形式です。

出力できる情報は次の 3 種類です。

- CTM ノード全体でまとめた情報

6. CTM で使用するコマンド

- キュー単位のスケジュールキュー情報
- メソッド単位のリクエスト情報

引数

-h

コマンドの使用方法が表示されます。

-CTMEdit < 編集出力形式 >[, < 編集出力形式 >...]

編集および出力する稼働統計情報の種類を指定します。

- nod : CTM ノード全体でまとめた情報を編集および出力します。
- que : キュー単位のスケジュールキュー情報を編集および出力します。
- mtd : メソッド単位のリクエスト情報を編集および出力します。

省略した場合は, nod , que , mtd のすべてが設定されます。

-CTMInterval < 時間間隔 >

稼働統計情報を編集および出力する時間の間隔 (単位: 分) を 1 ~ 1440 の整数で指定します。ctmstsstart コマンドの -CTMInterval オプションに指定した時間間隔の整数倍を指定してください。整数倍を指定しない場合は, 編集結果が正しく出力されません。

省略した場合は, "10" が設定されます。

-CTMTime [< 開始時刻 >][, < 終了時刻 >]

稼働統計情報を編集および出力する範囲を, データの取得時刻で指定します。

時分月日年を hhmm[MMDD[YYYY]] の形式で指定します。月 (MM), 日 (DD), および年 (YYYY) を省略すると, 現在の年月日が設定されます。

指定できる値の範囲を次の表に示します。

記号	意味	値の範囲	省略の可否
hh	時	00 hh 23	省略できません。
mm	分	00 mm 59	省略できません。
MM	月	01 MM 12	省略できます。
DD	日	01 DD 31	省略できます。
YYYY	年	1970 YYYY	省略できます。

開始時刻を省略した場合は, ファイルの先頭から終了時刻までが出力範囲になります。終了時刻を省略した場合は, 開始時刻からファイルの最後までが出力範囲になります。また, 開始時刻と終了時刻の両方を省略した場合は, 稼働統計情報ファイルのすべての情報が編集, 出力されます。

-CTMQueueName < キュー名 >[, < キュー名 >...]

稼働統計情報を編集および出力するキュー名を 1 ~ 63 文字の英数字, またはアンダースコア (_) で指定します。

このオプションは、-CTMEdit オプションに "que" を指定する場合、または -CTMEdit オプションを省略する場合に指定できます。

バッチアプリケーションを実行するシステムでは、キュー名としてスケジュールグループ名を指定してください。

-CTMMethod <メソッド名>[,<メソッド名>...]

稼働統計情報を編集および出力するメソッド名を指定します。

このオプションは、-CTMEdit オプションに "mtd" を指定する場合、または -CTMEdit オプションを省略する場合に指定できます。

-CTMCsv

編集および出力の結果を CSV 形式（コンマで区切ったテキスト形式）で出力します。

-CTMStdout

編集および出力の結果を標準出力に出力します。

表 6-2 引数（コマンドオプション）の組み合わせ一覧（ctmstsed コマンド）

コマンドオプション	-h	-CTMEdit			-CTMInterval	-CTMTIME	-CTMQueueName	-CTMMethod	-CTMCsv	-CTMStdout
		nod	que	mtd						
-h	-	-h	-h	-h	-h	-h	-h	-h	-h	-h
-CTMEdit	nod	-h	-				x	x		
	que	-h		-				x		
	mtd	-h			-		x			
-CTMInterval	-h				-					
-CTMTIME	-h					-				
-CTMQueueName	-h	x		x			-			
-CTMMethod	-h	x	x					-		
-CTMCsv	-h								-	
-CTMStdout	-h									-

（凡例）

: 同時に指定できます。

6. CTM で使用するコマンド

- : のあとに続くオプションが有効になります。
- x : 同時に指定できません。
- : 該当しません。

注

-h オプションとほかのオプションを指定する場合、-h が優先されます。

<稼働統計情報ファイル名> [<稼働統計情報ファイル名>...]

編集および出力する対象となる稼働統計情報ファイル名を指定します。半角スペースで区切って、最大 256 個のファイル名を指定できます。省略した場合は、標準入力から入力します。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

出力形式

- CTM ノード全体でまとめた情報を出力する形式 (-CTMEdit オプションに "nod" を指定)

```
*** Statistics Information by CTMNode ***
CTMDomain :aa..aa
CTMID      :bb..bb
-----
--
** Edit time cc..cc - dd..dd **
statistical data : ee..ee - ee..ee
no data          : ff..ff - ff..ff
-----
--
Event                Count      Maximum      Minimum      Average (unit)
Request Response Time(Reg) gg..gg      gg..gg      gg..gg      gg..gg (usec)
Request Time Out(Reg)   hh..hh      -            -            - (count)
Request Error(Reg)     ii..ii      -            -            - (count)
Request Response Time(OTM) jj..jj      jj..jj      jj..jj      jj..jj (usec)
Request Time Out(OTM)  kk..kk      -            -            - (count)
Request Error(OTM)    ll..ll      -            -            - (count)
Schedule Wait Count    mm..mm      mm..mm      mm..mm      mm..mm (count)
Schedule Wait Time     nn..nn      nn..nn      nn..nn      nn..nn (usec)
Request Overflow       oo..oo      -            -            - (count)
Regulator down        pp..pp      -            -            - (count)
OTMGateway down       qq..qq      -            -            - (count)
Regulator-Client      -           rr..rr      rr..rr      rr..rr (count)
OTMGateway-Client     -           ss..ss      ss..ss      ss..ss (count)
CTMDaemon-J2EEServer(Ctr) -           tt..tt      tt..tt      tt..tt (count)
CTMDaemon-J2EEServer(Req) -           uu..uu      uu..uu      uu..uu (count)
CTMDaemon-Regulator   -           vv..vv      vv..vv      vv..vv (count)
CTMDaemon-OTMGateway  -           ww..ww      ww..ww      ww..ww (count)
```

(凡例)

記号	説明
aa..aa	CTM ドメイン名称 (1 ~ 31 文字の英数字) です。
bb..bb	CTM 識別子 (1 ~ 31 文字の英数字) です。
cc..cc	編集開始時間です。
dd..dd	編集終了時間です。
ee..ee	編集時間内の稼働統計情報が取得されている時間帯です。
ff..ff	編集時間内で稼働統計情報が取得されていない時間帯です。
gg..gg	CTM レギュレータのレスポンス時間 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
hh..hh	CTM レギュレータのリクエストタイムアウト (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
ii..ii	CTM レギュレータのリクエスト障害 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
jj..jj	OTM ゲートウェイのレスポンス時間 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
kk..kk	OTM ゲートウェイのリクエストタイムアウト (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
ll..ll	OTM ゲートウェイのリクエスト障害 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
mm..mm	スケジュール待ち件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
nn..nn	スケジュール待ち時間 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
oo..oo	スケジュールできなかったリクエスト数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
pp..pp	CTM レギュレータ異常終了の件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
qq..qq	OTM ゲートウェイ異常終了の件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
rr..rr	CTM レギュレータとクライアント間のコネクション件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
ss..ss	OTM ゲートウェイと OTM クライアント間のコネクション件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
tt..tt	CTM デーモンと J2EE サーバ間の制御用のコネクション件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
uu..uu	CTM デーモンと J2EE サーバ間のリクエスト用のコネクション件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されま す
vv..vv	CTM デーモンと CTM レギュレータ間のコネクション件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。

6. CTM で使用するコマンド

記号	説明
ww..ww	CTM デーモンと OTM ゲートウェイ間のコネクション件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。

注

CTM デーモンと J2EE サーバ (バッチアプリケーションを実行するシステムの場合, バッチサーバ) 間制御用のコネクション件数は, J2EE サーバ数および J2EE サーバごとに生成しているキュー数の合計をカウントして出力します。

- CTM ノード全体でまとめた情報を CSV 形式で出力する形式 (-CTMEdit オプションに "nod" を指定し, かつ -CTMCsv オプションを指定)

```
*** Statistics Information by CTMNode ***
CTMDomain, CTMID, start, end, Event, Count, Maximum, Minimum, Average, unit
aa..aa, bb..bb, cc..cc, dd..dd, ee..ee, ff..ff, gg..gg, hh..hh, ii..ii, jj..jj
aa..aa, bb..bb, cc..cc, dd..dd, ee..ee, ff..ff, gg..gg, hh..hh, ii..ii, jj..jj
```

(凡例)

記号	説明
aa..aa	CTM ドメイン名称 (1 ~ 31 文字の英数字) です。
bb..bb	CTM 識別子 (1 ~ 31 文字の英数字) です。
cc..cc	編集開始時間です。
dd..dd	編集開始時間です。
ee..ee	イベント内容です。
ff..ff	発生件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
gg..gg	最大値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
hh..hh	最小値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
ii..ii	平均値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
jj..jj	単位です。

注

-CTMCsv オプションでは, 編集時間内の統計情報データが取得されている時間帯, および編集時間内で稼働統計が取得されていない時間帯は出力されません。

- キュー単位のスケジュールキュー情報を出力する形式 (-CTMEdit オプションに "que" を指定)

```

*** Statistics Information by Queue ***
CTMDomain :aa..aa
CTMID      :bb..bb
-----
--
QueueName : cc..cc
-----
--
** Edit time dd..dd - ee..ee **
statistical data : ff..ff - ff..ff
no data          : gg..gg - gg..gg
-----
--
Event                Count          Maximum      Minimum      Average (unit)
Schedule Wait Count  hh..hh        hh..hh      hh..hh      hh..hh (count)
Schedule Wait Time   ii..ii        ii..ii      ii..ii      ii..ii (usec)
Request Overflow     jj..jj        -           -           - (count)
CTMDaemon-J2EEServer (Req) -           kk..kk      kk..kk      kk..kk (count)

```

(凡例)

記号	説明
aa..aa	CTM ドメイン名称 (1 ~ 31 文字の英数字) です。
bb..bb	CTM 識別子 (1 ~ 31 文字の英数字) です。
cc..cc	J2EE アプリケーションの場合は、キュー名 (1 ~ 63 文字) です。 バッチアプリケーションの場合は、スケジュールグループ名です。
dd..dd	編集開始時間です。
ee..ee	編集終了時間です。
ff..ff	編集時間内の稼働統計情報が取得されている時間帯です。
gg..gg	編集時間内の稼働統計情報が取得されていない時間帯です。
hh..hh	スケジュール待ち件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。
ii..ii	スケジュール待ち時間 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。
jj..jj	スケジュールできなかったリクエスト数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。
kk..kk	CTM デーモンと J2EE サーバ間 (バッチアプリケーションを実行するシステムの場合はバッチサーバ間) のリクエスト用接続の件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。

注

QueueName : cc..cc 以下は、キュー単位に繰り返し出力されます。

- キュー単位のスケジュールキュー情報を CSV 形式で出力する形式 (-CTMedit オプションに "que" を指定し、かつ -CTMCsv オプションを指定)

6. CTM で使用するコマンド

```

*** Statistics Information by CTMQueue ***
CTMDomain , CTMID , start , end , Event , Count , Maximum , Minimum , Average , unit
aa..aa , bb..bb , cc..cc , dd..dd , ee..ee , ff..ff , gg..gg , hh..hh , ii..ii , jj..jj
aa..aa , bb..bb , cc..cc , dd..dd , ee..ee , ff..ff , gg..gg , hh..hh , ii..ii , jj..jj
:
:
:

```

(凡例)

記号	説明
aa..aa	CTM ドメイン名称 (1 ~ 31 文字の英数字) です。
bb..bb	CTM 識別子 (1 ~ 31 文字の英数字) です。
cc..cc	編集開始時間です。
dd..dd	編集開始時間です。
ee..ee	イベント内容です。
ff..ff	発生件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合 , "*****" が出力されます。
gg..gg	最大値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合 , "*****" が出力されます。
hh..hh	最小値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合 , "*****" が出力されます。
ii..ii	平均値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合 , "*****" が出力されます。
jj..jj	単位です。

注

-CTMCsv オプションでは、編集時間内の統計情報データが取得されている時間帯、および編集時間内で稼働統計が取得されていない時間帯は出力されません。

- メソッド単位のリクエスト情報を出力する形式 (-CTMEdit オプションに "mtd" を指定)

```

*** Statistics Information by Method ***
CTMDomain :aa..aa
CTMID      :bb..bb
-----
--
** Edit time cc..cc - dd..dd **
statistical data : ee..ee - ee..ee
no data         : ff..ff - ff..ff
-----
--
Interface      : gg..gg
Method         : hh..hh
-----
--
Event          Count          Maximum      Minimum      Average (unit)
Request Response Time (Reg) ii..ii      ii..ii      ii..ii      ii..ii (usec)
Request Time Out (Reg)    jj..jj      -           -           - (count)
Request Error (Reg)      kk..kk      -           -           - (count)
Request Response Time (OTM) ll..ll      11..11      11..11      11..11 (usec)
Request Time Out (OTM)   mm..mm      -           -           - (count)
Request Error (OTM)      nn..nn      -           -           - (count)

```

(凡例)

記号	説明
aa..aa	CTM ドメイン名称 (1 ~ 31 文字の英数字) です。
bb..bb	CTM 識別子 (1 ~ 31 文字の英数字) です。
cc..cc	編集開始時間です。
dd..dd	編集終了時間です。
ee..ee	編集時間内の稼働統計情報が取得されている時間帯です。
ff..ff	編集時間内の稼働統計情報が取得されている時間帯です。
gg..gg	インタフェース名 (1 ~ 31 文字の英数字) です。31 文字を超える場合は, "先頭から 15 文字" + "*" + "最後から 15 文字" の形式で出力されます。
hh..hh	メソッド名 (1 ~ 31 文字の英数字) です。31 文字を超える場合は, "先頭から 15 文字" + "*" + "最後から 15 文字" の形式で出力されます。
ii..ii	CTM レギュレータのレスポンス時間 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
jj..jj	CTM レギュレータのリクエストタイムアウト (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
kk..kk	CTM レギュレータのリクエスト障害 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
ll..ll	OTM ゲートウェイのレスポンス時間 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
mm..mm	OTM ゲートウェイのリクエストタイムアウト (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。
nn..nn	OTM ゲートウェイのリクエスト障害 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合, "*****" が出力されます。

6. CTM で使用するコマンド

注

Interface 以下の情報が、J2EE サーバのメソッド単位に繰り返し出力されます。

- メソッド単位のリクエスト情報を CSV 形式で出力する形式 (-CTMEdit オプションに "mtd" を指定し、かつ -CTMCsv オプションを指定)

```
*** Statistics Information by Method ***
CTMDomain, CTMID, start, end, Event, Count, Maximum, Minimum, Average, unit
aa..aa, bb..bb, cc..cc, dd..dd, ee..ee, ff..ff, gg..gg, hh..hh, ii..ii, jj..jj
aa..aa, bb..bb, cc..cc, dd..dd, ee..ee, ff..ff, gg..gg, hh..hh, ii..ii, jj..jj
:
```

(凡例)

記号	説明
aa..aa	CTM ドメイン名称 (1 ~ 31 文字の英数字) です。
bb..bb	CTM 識別子 (1 ~ 31 文字の英数字) です。
cc..cc	編集開始時間です。
dd..dd	編集開始時間です。
ee..ee	イベント内容です。
ff..ff	発生件数 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。
gg..gg	最大値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。
hh..hh	最小値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。
ii..ii	平均値 (11 けたの 10 進数) です。出力する値が 100000000000 以上の場合、"*****" が出力されます。
jj..jj	単位です。

注

-CTMCsv オプションでは、編集時間内の統計情報データが取得されている時間帯、および編集時間内で稼働統計が取得されていない時間帯は出力されません。バッチアプリケーションを実行するシステムの場合、メソッド単位のリクエスト情報は無効です。

ctmstsflush (バッファ内容の強制ファイル出力)

形式

```
ctmstsflush [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
```

機能

バッファリング中のメソッド単位のリクエスト情報を強制的にファイルに出力します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM ドメインマネージャが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

メソッド単位のリクエスト情報を強制的にファイルに出力させる、CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (.), またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、IP アドレスが CTM 識別子に設定されます。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

ctmstsstart (稼働統計情報の取得開始)

形式

```
ctmstsstart [-h] | [[-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
[-CTMInterval <時間間隔>] [-CTMCheckInterval <時間間隔>]]
```

機能

稼働統計情報ファイルへの情報の取得を開始します。

CTM デーモンを開始する前にこのコマンドを実行するとエラーになります。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM ドメインマネージャが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定

6. CTM で使用するコマンド

しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

稼働統計情報ファイルへの情報の取得を開始する、CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (.), またはアンダースコア (_) で指定します。

"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。マルチホームドホスト環境で使用する場合は、-CTMID オプションは省略できません。

省略した場合は、IP アドレスが CTM 識別子に設定されます。

-CTMInterval < 時間間隔 >

稼働統計情報ファイルに情報を取得する時間の間隔 (単位: 分) を 1 ~ 1440 の整数で指定します。省略した場合は "1" (分) が設定されます。

-CTMCheckInterval < 時間間隔 >

コネクション情報を取得する時間の間隔 (単位: 秒) を 1 ~ 60 の整数で指定します。コネクション数が多くなるとコネクション情報の取得に時間が掛かるので、コネクション数が多い場合は、指定する値を大きくしてください。省略した場合は "60" (秒) が設定されます。

戻り値

0:

正常終了しました。

0 以外:

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

注意事項

ctmstsstart コマンドを実行して -CTMInterval オプションに指定した時間が経過した時点で、稼働統計情報ファイルに情報が出力されるとは限りません。これは、ctmstsstart コマンドを実行して最初に XX 時 YY 分 00 秒 (X, Y: 符号なし整数) となった時から、稼働統計情報の取得が開始されるためです。この取得開始から -CTMInterval オプションに指定した時間が経過した時点で、稼働統計情報ファイルに情報が出力されます。

例えば、ctmstsstart コマンドに -CTMInterval 1 を指定して、18 時 10 分 30 秒に実行すると、稼働統計情報の取得は 18 時 11 分 00 秒に開始され、稼働統計情報ファイルが出力されるのは 18 時 12 分 00 秒となります。

ctmstsstop (稼働統計情報の取得終了)

形式

```
ctmstsstop [-h] [-CTMDomain <CTMドメイン名称>] [-CTMID <CTM識別子>]
```

機能

稼働統計情報ファイルへの情報の取得を終了します。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM ドメインマネージャが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

稼働統計情報ファイルへの情報の取得を停止する、CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (.), またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。マルチホームドホスト環境で使用する場合は、-CTMID オプションは省略できません。省略した場合は、IP アドレスが CTM 識別子に設定されます。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

注意事項

ctmstsstart コマンドの -CTMInterval オプションに指定した時間間隔の経過前に ctmstsstop コマンドを実行すると、ctmstsstop コマンドの実行までに取得した情報がすべてファイルに出力されます。

ctmtscgwd (OTM ゲートウェイの開始)

形式

```
ctmtscgwd [-h] |  
          [-CTMDomain <CTMドメイン名称>]  
          [-CTMID <CTM識別子>]  
          [-CTMPort <ポート番号>]  
          [-CTMMyHost <ホスト名称またはIPアドレス>]  
          [-CTMClientConnectCount <接続クライアントの最大数>]  
          [-CTMTPool]
```

機能

OTM クライアントのリクエストを CTM で処理するためのゲートウェイプロセスを開始します。OTM ゲートウェイは CTM デーモンが動作するホストと同じホストで開始してください。なお、OTM ゲートウェイを直接開始することはできません。

引数

-h

コマンドの使用方法が表示されます。

-CTMDomain <CTM ドメイン名称 >

CTM デーモンが属する CTM ドメイン名称を 1 ~ 31 文字の英数字、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる文字列は指定しないでください。省略した場合は "CTMDOMAIN" が設定されます。

-CTMID <CTM 識別子 >

CTM デーモンの CTM 識別子を 1 ~ 31 文字の英数字、ピリオド (.)、またはアンダースコア (_) で指定します。"CTM" または "ctm" で始まる CTM 識別子は指定しないでください。ピリオド (.) は、IP アドレスを指定する場合だけ使用できます。省略した場合は、IP アドレスが CTM 識別子に設定されます。例えば、ホストの IP アドレスが "172.17.112.43" のとき、"172.17.112.43" という文字列が CTM 識別子となります。

-CTMPort <ポート番号 >

OTM クライアントが OTM ゲートウェイの新しいコネクションを探すときに使用するポート番号を 5001 ~ 65535 の整数で指定します。省略した場合は、OS が自動的に割り当てたポート番号を使用します。

-CTMMyHost <ホスト名称または IP アドレス >

マルチホームまたは IP アドレスを引き継ぐ系切り替え環境で CTM が使用するホスト名称または IP アドレスを 1 ~ 64 文字の文字列で指定します。省略した場合は、hostname コマンドで取得されるホスト名称が設定されます。

-CTMID オプションを省略して -CTMMyHost オプションを指定した場合、デフォ

ルトの CTM 識別子は -CTMMyHost オプションで指定した IP アドレスになります。

-CTMClientConnectCount < 接続クライアントの最大数 >

OTM ゲートウェイに接続できるコネクション数を 1 ~ 32767 の整数で指定します。コネクション数は、OTM クライアントからの接続数です。省略した場合は "64" が設定されます。

-CTMTPool

OTM ゲートウェイをスレッドプーリング (ThreadPool) モデルで動作させます。省略した場合は、スレッドパーセション (ThreadSession) で動作させます。スレッドプーリングおよびスレッドパーセションの詳細については、マニュアル「Borland(R) Enterprise Server VisiBroker(R) デベロッパーズガイド」を参照してください。

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

注意事項

OTM ゲートウェイは開始後にカレントディレクトリを移動します。そのため、次に示す場合のコアファイルの出力先は、実行形式ファイルのディレクトリとは異なります。

- "ctmdmstop -CTMForce -CTMDebugInfo", kill コマンドなどで OTM ゲートウェイを強制終了した場合
- OTM ゲートウェイが異常終了した場合

移動後のカレントディレクトリを次に示します。

- Windows の場合
"%CTMSPOOL%\¥adm¥CTM 識別子 ¥tgwd¥ カレントディレクトリ ID¥"
- UNIX の場合
"\$CTMSPOOL/¥adm/CTM 識別子 /¥tgwd/ カレントディレクトリ ID/"
カレントディレクトリ ID は、メッセージ KFCT70606-I を参照してください。

ctmver (実行形式ファイルおよびライブラリのバージョン情報の出力)

形式

ctmver -h | <ファイルパス>

6. CTM で使用するコマンド

機能

CTM が提供する実行形式ファイルおよびライブラリのバージョン情報を出力します。

引数

-h

コマンドの使用方法が表示されます。

<ファイルパス>

対象となるファイルパス（相対パスまたは絶対パス）を指定します。

出力形式

```
Information For : aa...aa
Product Name : bb...bb
PP Number : cc...cc
Version : dd...dd
Copyright : ee...ee
Company : ff...ff
Build OS : gg...gg
Build Compiler : hh...hh
Build Date : ii...ii jj...jj
Build Number : kk...kk
```

(凡例)

aa...aa : ファイル名称 (文字列)
bb...bb : 製品名称 (文字列)
cc...cc : PP 型名 (文字列)
dd...dd : バージョンおよびリビジョン (文字列)
ee...ee : コピーライト (文字列)
ff...ff : 会社名 (文字列)
gg...gg : ビルド OS (文字列)
hh...hh : ビルドコンパイラ (文字列)
ii...ii : ビルド日付 (文字列)
jj...jj : ビルド時刻 (文字列)
kk...kk : 保守情報 (文字列)

戻り値

0 :

正常終了しました。

0 以外 :

コマンド処理中にエラーが発生したため、異常終了しました。出力されるメッセージに従って対策したあと、再度コマンドを実行してください。

7

Management Server で使用するコマンド

この章では、Management Server で使用するコマンドの入力形式、機能などについて説明します。

7.1 Management Server で使用するコマンドの一覧

7.2 Management Server で使用するコマンドの詳細

7.3 mngsvrutil コマンドのサブコマンドの詳細

7.1 Management Server で使用するコマンドの一覧

Management Server で使用するコマンドの一覧を、次の表に示します。

表 7-1 Management Server で使用するコマンドの一覧

コマンド名称	分類	概要
adminagentcheck	運用管理エージェントの稼働確認	運用管理エージェントが稼働しているかどうかを確認します。
adminagentctl	運用管理エージェントの起動と停止	運用管理エージェントのサービスを起動または停止します。
daemon	デーモンプロセスの起動	Management Server, 運用管理エージェントをデーモンプロセスとして起動します。 このコマンドは、UNIX 用です。
kill	プロセスの強制停止	プロセスを強制終了します。 このコマンドは、Windows 用です。
mngautorun	サービスの設定	Management Server および運用管理エージェントのサービスの設定を変更します。 このコマンドは、Windows 用です。
mngenvsetup	管理グループの設定	OS のグループを Manager Sever 管理グループ、統合ユーザ管理グループに設定します。 このコマンドは、UNIX 用です。
mngsvr_adapter_setup	Cosminexus アダプタコマンドのセットアップとアンセットアップ	JP1/IM 連携で使用する Cosminexus アダプタコマンドのセットアップまたはアンセットアップを実行します。
mngsvr_monitor_setup	Cosminexus モニタ起動コマンドのセットアップとアンセットアップ	JP1/IM 連携で使用する Cosminexus モニタ起動コマンドのセットアップまたはアンセットアップを実行します。 このコマンドは、Windows 用です。
mngsvrctl	Management Server の起動 / 停止 / セットアップ	Management Server を起動または停止します。 または、Management Server の実行に必要なセットアップをします。
mngsvrupdate	Management Server の移行コマンド	Management Server で管理している情報を、最新のバージョンで使用する形式に移行します。
mngsvrutil	Management Server の運用管理コマンド	Management Server に接続して、運用管理ドメイン内での操作を実行します。
mstrexport	Management Server 管理ファイルの退避	Management Server 管理ファイルを収集して ZIP 形式で出力します。
mstrimport	Management Server 管理ファイルの回復	回復に成功したファイルのパス名を、システムプロパティを含んだ形式で標準出力に出力します。

7.2 Management Server で使用するコマンドの詳細

Management Server で使用する各コマンドの入力形式、機能などを次に示します。

コマンドの格納先

Management Server サーバで使用するコマンドは、次のディレクトリに格納されています。

Windows の場合

- mngsvr_monitor_setup コマンド以外のコマンド
`<Cosminexus のインストールディレクトリ >%manager%bin%`
 - mngsvr_monitor_setup コマンド
`<Cosminexus のインストールディレクトリ >%manager%externals%jpl%mngsvrmonitor%`
- なお、このコマンドは、運用管理クライアントマシンのコピー先で実行します。

UNIX の場合

`/opt/Cosminexus/manager/bin/`

adminagentcheck (運用管理エージェントの稼働確認)

形式

他ホストの運用管理エージェントの稼働状況を確認する場合

```
adminagentcheck [-t <タイムアウト時間>] [<ホスト名>[:<ポート番号>]]
```

自ホストの運用管理エージェントの稼働状況を確認する場合

```
adminagentcheck [-t <タイムアウト時間>]
```

機能

運用管理エージェントが稼働しているかどうかを、Management Server を経由しないで確認します。

引数

`-t <タイムアウト時間>`

運用管理エージェントとの接続タイムアウト時間 (単位: 秒) を、0 ~ 2147483 の整数で指定します。0 を指定した場合、タイムアウトは無限と解釈されます。その場合、接続が確立されるまで、またはエラーが発生するまで待ち続けます。このオプションを省略した場合、タイムアウト時間は 5 (秒) です。

7. Management Server で使用するコマンド

< ホスト名 >

運用管理エージェントとの通信で使用するドット記法での IP アドレス (xxx.xxx.xxx.xxx), または名前解決できるホスト名を指定します。なお, xxx には 0 ~ 255 の整数を指定します。

< ポート番号 >

運用管理エージェントとの通信で使用するポート番号を 1 ~ 65535 の整数で指定します。

オプションが省略された場合に, adminagent.properties に省略値が設定されているときは, その値が使用されます。値が設定されていない場合, またはポート番号に指定できない値が設定されている場合はデフォルト値が使用されます。オプションの省略時に使用される adminagent.properties のプロパティ名, デフォルト値を次に示します。

オプション	プロパティ名	デフォルト値
ホスト名	adminagent.adapter.bind_host	127.0.0.1
ポート番号	adminagent.adapter.port	20295

戻り値

0 :

運用管理エージェントは稼働しています。

1 :

コマンド実行時エラーです。

指定されたホスト名がアドレス解決できない場合, 1 となります。

2 :

コマンドシンタックスエラーです。

103 :

運用管理エージェントは稼働していません。

すべての通信エラーおよびタイムアウト時に, 103 となります。

注意事項

オプションが省略された場合は, adminagent.properties のポート番号に指定範囲外の値が設定されていてもデフォルト値が設定されますが, オプションに指定範囲外の値が設定された場合は, シンタックスエラーとなります。

adminagentcheck コマンドは, 運用監視を行うバッチまたはシェルからの使用が想定されています。そのため, 標準出力およびエラー出力には何も出力されません。運用管理エージェントが稼働しているかどうか, およびコマンドエラーについては, 戻り値から判定してください。ただし, コマンドシンタックスエラーの場合だけ, Usage が出力されます。

adminagentctl (運用管理エージェントの起動と停止)

形式

```
adminagentctl {start|stop}
```

機能

運用管理エージェントのサービスを起動または停止します。

UNIX の場合に、運用管理エージェントをデーモンプロセスとして起動する方法については、「daemon (デーモンプロセスの起動)」を参照してください。

このコマンドは、Administrator 権限 (Windows の場合)、または root 権限 (UNIX の場合) のあるユーザが実行してください。ただし、UNIX の場合、root 権限がない特定のユーザにコマンドの実行権限を設定することもできます。設定については、「mngenvsetup (管理グループの設定)」を参照してください。

引数

引数の指定がない場合、コマンドの使用方法を表示します。

start

運用管理エージェントのサービスを開始します。

stop

運用管理エージェントのサービスを終了します。

戻り値

0 :

正常終了しました。

1 :

実行時エラーが発生しました。

2 :

シンタックスエラーが発生しました。

入力例

```
adminagentctl start
```

daemon (デーモンプロセスの起動)

形式

Management Server を起動する場合

Management Server をデーモンプロセスとして起動する場合、次の方法で起動して

7. Management Server で使用するコマンド

ください。

```
/opt/Cosminexus/manager/bin/daemon -cf /opt/Cosminexus/manager/bin/mngsvrctl start
```

mngsvrctl (Management Server の起動 / 停止 / セットアップ) の詳細については、「mngsvrctl (Management Server の起動 / 停止 / セットアップ)」を参照してください。

運用管理エージェントを起動する場合

運用管理エージェントをデーモンプロセスとして起動する場合、次の方法で起動してください。

```
/opt/Cosminexus/manager/bin/daemon -cf /opt/Cosminexus/manager/bin/adminagentctl start
```

adminagentctl (運用管理エージェントの起動と停止) の詳細については、「adminagentctl (運用管理エージェントの起動と停止)」を参照してください。

機能

Management Server, 運用管理エージェントを, デーモンプロセスとして起動します。なお, このコマンドは, UNIX 用です。

kill (プロセスの強制停止)

形式

```
kill <プロセスID> [<プロセスID> ...]
```

機能

プロセスを強制終了します。運用管理ポータルから論理サーバを停止できなくなった場合などに実行します。

このコマンドは, Administrator 権限のあるユーザが実行してください。なお, このコマンドは, Windows 用です。

引数

<プロセス ID>

強制終了するプロセスの ID を指定します。

複数指定できます。

入力例

```
kill 2034
```

mngautorun (サービスの設定)

形式

```
mngautorun [-d] [{server|agent|both}]
```

機能

Management Server および運用管理エージェントのサービスの設定を変更します。このコマンドでは、次の設定をします。

- マシン起動と同時に、Management Server および運用管理エージェントのサービスを起動するように設定します。
- マシン起動時に、Management Server および運用管理エージェントのサービスの両方が起動する場合、運用管理エージェント起動後に Management Server を起動するように依存関係を設定します。

なお、このコマンドは、Windows 用です。

引数

オプションの指定がない場合、コマンドの使用方法が表示されます。

-d

Management Server および運用管理エージェントのサービスに関する設定をインストール時の状態に戻します。

server

Management Server のサービスのスタートアップに関する設定を自動に変更します。

agent

運用管理エージェントのサービスのスタートアップに関する設定を自動に変更します。

both

Management Server および運用管理エージェントのサービスのスタートアップに関する設定を自動に変更して、運用管理エージェント起動後に Management Server を起動するように依存関係を設定します。

入力例

入力例 1

Management Server サービスと運用管理エージェントサービスを OS 起動時に開始する場合

```
mngautorun both
```

入力例 2

7. Management Server で使用するコマンド

インストール時の状態に戻す場合

```
mngautorun -d
```

注意事項

このコマンドを使用して Management Server，および運用管理エージェントのサービスの設定を変更した環境に，アプリケーションサーバを上書きインストール，またはアップグレードインストールすると，サービスの設定がリセットされます。その場合は，再度このコマンドを実行してください。

mngenvsetup (管理グループの設定)

形式

```
mngenvsetup [-mng | -ua] {<OSのグループ名> | -d}
```

機能

root 権限を持つユーザしかできなかった Management Server で使用するコマンドおよび統合管理ユーザで使用するコマンドを，root 権限を持たない特定のユーザが実行できるように設定できます。なお，このコマンドは，UNIX 用です。

Management Server で使用するコマンドを実行できるユーザのグループを Management Server 管理グループといいます。統合管理ユーザで使用するコマンドを実行できるユーザのグループを統合ユーザ管理グループといいます。

このコマンドは，OS のグループを Management Server 管理グループと統合ユーザ管理グループとして設定します。

このコマンドの対象となるコマンドを次に示します。

root 権限以外の運用で実行できるコマンド	管理グループ	
	Management Server 管理グループ	統合ユーザ管理グループ
adminagentctl		-
convpw	-	
mngsvr_adapter_setup		-
mngsvrctl		-
ssoexport	-	
ssogenkey	-	
ssoimport	-	
uachpw	-	

(凡例)

- ：該当します。
- ：該当しません。

注 統合ユーザ管理グループのコマンドの詳細については、「13. 統合ユーザ管理で使用するコマンド」を参照してください。

引数の指定と管理グループの設定は次のとおりです。

- `-mng` を指定すると Management Server 管理グループだけを設定します。
- `-ua` を指定すると統合ユーザ管理グループだけを設定します。
- `-mng`、`-ua` を指定しないで、OS のグループ名を指定した場合は、Management Server 管理グループおよび統合ユーザ管理グループの両方に、指定した OS のグループが設定されます。
- `-d` だけを指定すると、Management Server 管理グループおよび統合ユーザ管理グループの両方の設定が削除されます。コマンドを実行できるのは、`root` 権限を持つユーザだけに戻ります。また、Management Server 管理グループおよび統合ユーザ管理グループは、アプリケーションサーバのインストール時の OS グループに戻ります。
- `-d` と `-mng` または `-ua` を同時に指定すると、Management Server 管理グループまたは統合ユーザ管理グループのどちらかの設定が削除されます。設定が削除された管理グループのコマンドを実行できるのは、`root` 権限を持つユーザだけに戻ります。

Management Server 管理グループまたは統合ユーザ管理グループが設定された状態でこのコマンドを実行すると上書き設定されます。

引数

`-mng` <OS のグループ名 >

OS のグループを Management Server 管理グループに設定します。OS のグループ名を省略した場合には USAGE が表示されます。

`-ua` <OS のグループ名 >

OS のグループを統合ユーザ管理グループに設定します。OS のグループ名を省略した場合には USAGE が表示されます。

`-d`

Management Server 管理グループおよび統合ユーザ管理グループの設定が削除されます。コマンドを実行できるのは、`root` 権限を持つユーザだけに戻ります。

入力例

Management Server 管理グループおよび統合ユーザ管理グループを `manager` に設定する場合

```
mngenvsetup manager
```

Management Server 管理グループを `manager` に設定する場合

```
mngenvsetup -mng manager
```

7. Management Server で使用するコマンド

統合ユーザ管理グループを manager に設定する場合

```
mngenvsetup -ua manager
```

Management Server 管理グループおよび統合ユーザ管理グループを root に戻す場合

```
mngenvsetup -d
```

注意事項

このコマンドを実行すると、Management Server で使用するコマンドまたは統合ユーザ管理で使用するコマンドに SUID ビットが付加されます。このため、アクセス権を与えるユーザを適切に設定する必要があります。

mngsvr_adapter_setup (Cosminexus アダプタコマンドのセットアップとアンセットアップ)

形式

```
mngsvr_adapter_setup [ -i | -u ] [ -t { IM_CS | IM_CM } ]
```

機能

JP1/IM 連携をする場合に、Cosminexus アダプタコマンドをセットアップまたはアンセットアップします。コマンドを実行すると、JP1/Base の環境に Cosminexus 用のアダプタコマンド設定ファイルを追加します。

このコマンドは、アプリケーションサーバの運用管理サーバ (Management Server を配置しているホスト) で実行します。

このコマンドは、Administrator 権限 (Windows の場合)、または root 権限 (UNIX の場合) があるユーザが実行してください。ただし、UNIX の場合、root 権限がない特定のユーザにコマンドの実行権限を設定することもできます。設定については、「mngenvsetup (管理グループの設定)」を参照してください。

引数

引数の指定がない場合、JP1/IM - Central Scope 連携用のアダプタコマンドのセットアップを実行します。

-i

セットアップを実行します。

-u

アンセットアップを実行します。

-t IM_CS

JP1/IM - Central Scope 連携用のアダプタコマンドのセットアップを実行します。

-t IM_CM

JP1/IM・CM 連携用のアダプタコマンドのセットアップを実行します。

入力例

```
mngsvr_adapter_setup -i -t IM_CM
```

mngsvr_monitor_setup (Cosminexus モニタ起動コマンドのセットアップとアンセットアップ)

形式

```
mngsvr_monitor_setup [ -i | -u ]
```

機能

JP1/IM 連携をする場合に、Cosminexus モニタ起動コマンドをセットアップまたはアンセットアップします。

このコマンドは、<Cosminexus のインストールディレクトリ>¥manager¥externals¥jp1¥mngsvrmonitor 下に格納されています。運用管理クライアントマシンのコピー先で実行します。

このコマンドは、Administrator 権限があるユーザが実行してください。なお、このコマンドは、Windows 用です。

引数

引数の指定がない場合は、セットアップを実行します。

-i

セットアップを実行します。

-u

アンセットアップを実行します。

mngsvrctl (Management Server の起動 / 停止 / セットアップ)

形式

Management Server を起動する場合

Windows の場合

```
mngsvrctl start
```

UNIX の場合

```
/opt/Cosminexus/manager/bin/mngsvrctl start
```

Management Server を停止する場合

7. Management Server で使用するコマンド

Windows の場合

```
mngsvrctl stop
```

UNIX の場合

```
/opt/Cosminexus/manager/bin/mngsvrctl stop
```

Management Server をセットアップする場合

Windows の場合

```
mngsvrctl setup [<サーバ名>] [-u <管理ユーザID>] [-p <管理ユーザパスワード>] [-np <変更後の管理ユーザパスワード>] [-nu <変更後のユーザID>]
```

UNIX の場合

```
/opt/Cosminexus/manager/bin/mngsvrctl setup [<サーバ名>] [-u <管理ユーザID>] [-p <管理ユーザパスワード>] [-np <変更後のユーザパスワード>] [-nu <変更後のユーザID>]
```

機能

Management Server を起動または停止します。または、アプリケーションサーバを初めてインストールしたマシンで、Management Server を実行するために必要なセットアップをします。

管理ユーザアカウントの省略機能を無効にしている場合、セットアップと同時に、Management Server へログインするための管理ユーザアカウント（管理ユーザ ID および管理ユーザパスワード）を設定できます。セットアップ時に管理ユーザアカウントを設定しないと、Management Server の起動時および停止時に管理ユーザアカウントの設定が必要になります。また、管理ユーザアカウントがすでに設定されている場合は、別のユーザアカウントを指定することで管理ユーザアカウントを変更することもできます。

管理ユーザアカウントの省略機能を有効にしている場合、このコマンドの実行時および Management Server の起動後に管理ユーザアカウントを設定する必要はありません。

管理ユーザアカウントの省略機能については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の管理ユーザアカウントの設定を省略する場合に関する説明を参照してください。

このコマンドは、Administrator 権限（Windows の場合）、または root 権限（UNIX の場合）のあるユーザが実行してください。ただし、UNIX の場合、root 権限がない特定のユーザにコマンドの実行権限を設定することもできます。設定については、「mngenvsetup（管理グループの設定）」を参照してください。

UNIX の場合に、Management Server をデーモンプロセスとして起動する方法については、「daemon（デーモンプロセスの起動）」を参照してください。

引数

引数の指定がない場合、コマンドの使用方法が表示されます。

start

Management Server を起動します。

stop

Management Server を停止します。

setup

Management Server をセットアップします。

< サーバ名 >

Management Server の名称 (Web コンテナのサーバ名称) を指定します。省略した場合は、「cosmi_m」が仮定されます。

-u < 管理ユーザ ID >

Management Server の管理ユーザ ID を指定します。

-p < 管理ユーザパスワード >

Management Server の管理ユーザパスワードを指定します。

-np < 変更後の管理ユーザパスワード >

新しい Management Server の管理ユーザパスワードを指定します。

-nu < 変更後の管理ユーザ ID >

新しい Management Server の管理ユーザ ID を指定します。

設定済みの管理ユーザパスワードを解除する場合は、空文字 ("") を指定します。

戻り値

0 :

正常終了しました。

1 :

実行時エラーが発生しました。

2 :

シンタックスエラーが発生しました。

入力例

Management Server をセットアップする場合

```
mngsvrctl setup [<サーバ名>]
```

セットアップ時に管理ユーザアカウントの設定もする場合

```
mngsvrctl setup [<サーバ名>] -u admin01 -p p1574809
```

設定済みの管理ユーザアカウントを変更する場合

```
mngsvrctl setup -u admin01 -p p1574809 -np passwd123 -nu admin02
```

管理ユーザパスワードを解除する場合

7. Management Server で使用するコマンド

```
mngsvrctl setup localhost -u admin01 -p p1574809 -np ""
```

注意事項

UNIX の場合の mngsvrctl の注意事項を次に示します。

Management Server を起動する前に、umask 022 を設定してください。

コマンドから Management Server を起動した場合、umask に影響されるファイルは、それぞれ次の owner および group で作成します。

- AIX の場合
root/system
- HP-UX の場合
root/sys
- Linux の場合
root/root
- Solaris の場合
root/other

umask に影響されるファイルを次に示します。

- 構成情報の退避 / 回復画面で退避した構成情報ファイル (ユーザ任意)
- J2EE アプリケーションまたはリソースのインポート時に作成される J2EE アプリケーションファイル (EAR ファイル), またはリソース情報ファイル (mapps_< 論理サーバ名 >_< 内部 ID >)
- 登録された J2EE アプリケーション (< アプリケーション表示名 >_< 登録時刻 >)
- ssogenkey コマンド実行時またはリポジトリ管理で暗号鍵ファイルの設定時に作成される暗号化 / 復号化用の鍵ファイル (ユーザ任意)
- J2EE サーバ, またはバッチサーバへの設定情報の配布時に作成される運用監視エージェントの設定ファイル (mngagent.< 論理サーバ名 >.properties)
- snapshot ログ収集の直前に作成される snapshot ログ機能で設定されている環境変数の格納ファイル (envinfo.txt)
- J2EE サーバ, またはバッチサーバへの設定情報の配布時に作成される J2EE サーバ, またはバッチサーバのユーザ定義ファイルのバックアップファイル (usrconf.cfg.bak)
- J2EE サーバ, またはバッチサーバへの設定情報の配布時に作成される J2EE サーバ, またはバッチサーバのユーザ定義ファイルのバックアップファイル (usrconf.properties.bak)

Management Server を起動する場合、ロケールを日本語に設定してから実行してください。

ksh を使用している環境でロケールを日本語に設定する場合の例を次に示します。

- AIX の場合の設定例
export LANG=Ja_JP または export LANG=ja_JP
- HP-UX の場合の設定例
export LANG=ja_JP.SJIS または export LANG=ja_JP.eucJP

- Linux の場合の設定例

```
export LANG=ja_JP.utf8
```
- Solaris の場合の設定例

```
export LANG=Ja_JP.PCK または export LANG=ja_JP.eucJP
```

Management Server をセットアップする場合の注意事項を次に示します。

管理ユーザアカウントの省略機能を有効にしてこのコマンドを実行した場合、共通引数の `-u` オプション、`-p` オプション、`-np` オプション、および `-nu` オプションの設定は無効になります。このときメッセージが出力され、セットアップ処理は続行されます。

管理ユーザアカウントの省略機能を無効にして管理ユーザ ID が設定されていない状態で `-nu` オプションを指定したコマンドを実行すると、`-u` オプションの指定は無視されて、`-nu` オプションで指定した管理ユーザ ID が設定されます。また、管理ユーザアカウントの省略機能を無効にして管理ユーザパスワードが設定されていない状態で `-np` オプションを指定したコマンドを実行すると、`-p` オプションの指定は無視されて、`-np` オプションで指定した管理ユーザパスワードが設定されます。

mngsvrupdate (Management Server の移行コマンド)

形式

```
mngsvrupdate [-t <ターゲット名称>] [-k <ターゲット種別>]
               [<サブコマンド> [<引数>]]
```

機能

Cosminexus Component Container のバージョンアップに合わせて、Management Server で管理する情報を最新のバージョンで使用する形式に移行します。

このコマンドは、Management Server が動作するホストで実行してください。

なお、移行手順については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

引数

`-t` <ターゲット名称>

サブコマンドの対象となるターゲット名称（論理サーバ名またはホスト名）を指定します。

運用管理ポータル「論理サーバの構成定義」の定義に従います。

省略した場合、すべてのホストのすべての論理サーバが指定されます。

`-k` <ターゲット種別>

サブコマンドの対象となるターゲットの種別を指定します。次のどちらかを指定します。

7. Management Server で使用するコマンド

- logicalServer : 論理サーバ
- host : ホスト

省略した場合、または上記以外の値を指定した場合は、logicalServer が設定されません。

<サブコマンド> <引数>

指定しない場合は Usage が表示されます。

指定できるサブコマンドと引数について説明します。

d2r [< データソースの表示名 > [< リソースアダプタの表示名 >]]

Management Server で管理するデータソースの情報をリソースアダプタに移行します。移行する対象は、< ターゲット名称 > および < ターゲット種別 > で指定します。

- 引数に < データソースの表示名 > を指定した場合
指定したデータソースだけが移行されます。この引数を指定した場合、< ターゲット名称 > に論理サーバ名を指定する必要があります。また、この引数を省略した場合、該当する論理サーバにインポートされたすべてのデータソースが移行対象となります。
- 引数に < リソースアダプタの表示名 > を指定した場合
移行後のデータソースの表示名は、この引数で指定した値になります。ただし、すでに存在するリソースアダプタが指定された場合は移行されません。また、この引数を省略した場合、データソースの表示名がリソースアダプタの表示名になります。

移行処理では、次のファイルを編集します。

- Windows の場合
<Cosminexus のインストールディレクトリ >%manager%config%mmodel.xml
- UNIX の場合
/opt/Cosminexus/manager/config/mmodel.xml

なお、mmodel.xml は mmodel.xml.bak にバックアップしますが、すでに mmodel.xml.bak が存在する場合は上書きされます。

入力例

運用管理ドメイン内のすべてのホストを一括して移行する場合

```
mngsvrupdate d2r
```

ホストを指定して移行する場合

```
mngsvrupdate -t hostA -k host d2r
```

論理サーバを指定して移行する場合

```
mngsvrupdate -t MyServer d2r ds ds1
```

注意事項

このコマンドは、Management Server が停止していることを確認してから実行してください。Management Server の起動中に実行すると、変更が反映されない場合があります。

移行コマンドを同時に実行すると正しく環境が移行されないことがあります。コマンドは同時に実行しないでください。

移行コマンドを実行する前に、次のファイルをほかの場所に退避してください。

- Windows の場合

```
<Cosminexus のインストールディレクトリ >%manager%config%model.xml
```

- UNIX の場合

```
/opt/Cosminexus/manager/config/mmodel.xml
```

mngsvrutil (Management Server の運用管理コマンド)

形式

```
mngsvrutil [-m <ホスト名> [:<ポート番号>]
            [-u <管理ユーザID>] [-p <管理ユーザパスワード>]
            [[-o <ファイル名>] [-h] [-f <出力形式>]
            [-t <ターゲット名称>] [-k <ターゲット種別>]
            [-s] [-i <監視時間間隔>] [-l <最大監視時間>]
            [-w <稼働確認待ち時間>] <サブコマンド> <引数>]]
```

機能

Management Server に接続して、運用管理ドメインについての操作ができます。また、JP1/Cm2/ESA などの SNMP エージェント製品を使用して、mngsvrutil コマンドの出力結果を MIB に変換すると、SNMP Manager 製品と連携できます。

このコマンドでは、次の機能を使用できます。

- 運用管理ドメインの構成情報の取得
運用管理ドメインの名称、および運用管理ドメインを構成する論理サーバの一覧を取得して、表示します。
- 論理サーバのステータス情報の取得
運用管理ドメインを構成する論理サーバの一覧に、ステータス（稼働しているかどうか）情報を付加して表示します。
- 論理サーバにデプロイされている J2EE アプリケーションの情報の取得
J2EE アプリケーションの一覧を取得して、表示します。
- 論理サーバの性能情報の取得、および統計情報についての設定
論理サーバの性能情報（JavaVM、トランザクション、HTTP サーバ、HTTP サーバコネクタなど）を取得します。また、統計情報のサンプリング時間を設定します。
- 論理サーバの起動または停止
論理サーバを起動または停止します。

7. Management Server で使用するコマンド

- 論理サーバの設定ファイルの再読み込み
論理サーバを再起動しないで、論理サーバで使用する設定ファイルを再読み込みします。また、論理 J2EE サーバ、または論理 SFO サーバで、接続先ホストの設定ファイルを再読み込みすることもできます。
- 論理サーバの追加・削除
論理サーバを追加または削除します。
- 論理サーバのアプリケーション管理
J2EE アプリケーション / J2EE リソースの開始 / 停止および一覧表示をします。アプリケーション管理の動作については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」を参照してください。
- 性能解析トレースの収集
論理パフォーマンストレーサから性能解析トレースを収集します。
- CTM キューの閉塞 / 閉塞解除
CTM のスケジュールキューの閉塞、および閉塞の解除をします。
- CTM 稼働統計情報の収集
CTM から稼働統計情報を収集します。
- 同時実行スレッド数の動的変更
Web アプリケーションの同時実行スレッド数を動的に変更します。

引数

-m < ホスト名 > [: < ポート番号 >]

Management Server のホスト名とポート番号を指定します。

- < ホスト名 >

ドット記法での IP アドレス指定 (xxx.xxx.xxx.xxx), または名前解決できるホスト名で指定します。なお, xxx には 0 ~ 255 の整数を指定します。

- < ポート番号 >

Management Server の設定に従います。デフォルト値は 28080 です。

-u < 管理ユーザ ID >

mngsvrctl setup コマンドまたは運用管理ポータル「管理ユーザアカウントの設定」で設定した管理ユーザ ID を指定します。

運用管理ポータル「管理ユーザアカウントの設定」で入力できる文字に従います。管理ユーザアカウントの省略機能が有効の場合、管理ユーザ ID を指定しても無効となります。管理ユーザアカウントの省略機能については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の管理ユーザアカウントの設定を省略する場合に関する説明を参照してください。

-p < 管理ユーザパスワード >

mngsvrctl setup コマンドまたは運用管理ポータル「管理ユーザアカウントの設定」で設定した管理ユーザ ID に対応する管理ユーザパスワードを指定します。

運用管理ポータル「管理ユーザアカウントの設定」で入力できる文字に従います。

なお、パスワードを設定していない場合、`-p` オプションを指定する必要はありません。

管理ユーザアカウントの省略機能が有効の場合、管理ユーザパスワードを指定しても無効となります。管理ユーザアカウントの省略機能については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の管理ユーザアカウントの設定を省略する場合に関する説明を参照してください。

-o <ファイル名>

出力先のファイル名を指定します。指定されたファイルに上書き出力します。なお、指定できるファイル名は、コマンドを実行する OS に依存します。また、省略した場合、標準出力に出力します。

-h

ヘッダを出力しない場合に指定します。

-f <出力形式>

接続する Management Server で提供している出力形式を指定します。

Management Server では、次に示す出力形式を提供しています。

- `csv` : CSV 形式 (デフォルト値)
- `snmp` : SNMP 連携用形式

-t <ターゲット名称>

サブコマンドの対象となるターゲット名称 (論理サーバ名またはホスト名) を指定します。

運用管理ポータルの「論理サーバの構成定義」の定義に従います。

-k <ターゲット種別>

サブコマンドの対象となるターゲットの種別を指定します。次のどちらかを指定します。

- `logicalServer` : 論理サーバ
- `host` : ホスト

省略した場合、または上記以外の値を指定した場合は、`logicalServer` が設定されます。

-s

論理サーバや J2EE アプリケーションの起動または停止の完了を待つ場合に指定します。

このオプションは、サブコマンドに "start" または "stop" を指定した場合に有効です。

なお、このオプションを指定した場合、起動 / 停止障害発生時、障害回復のためにサブコマンドのリトライ処理が実行されます。

-i <監視時間間隔>

論理サーバや J2EE アプリケーションの起動・停止の完了を監視する際の監視時間

7. Management Server で使用するコマンド

間隔を、0 ~ 2147483647 の整数（単位：秒）で指定します。省略時のデフォルトは 3 秒です。

なお、このオプションは、起動 / 停止障害発生時のサブコマンドのリトライ処理間隔としても利用されます。

-l <最大監視時間>

論理サーバや J2EE アプリケーションの起動・停止の完了を監視する際の最大監視時間を、0 ~ 2147483647 の整数（単位：秒）で指定します。

指定した値の時間を超えても論理サーバや J2EE アプリケーションが起動・停止しない場合、コマンドは異常終了します。省略時のデフォルトは 60 秒です。

-w <稼働確認待ち時間>

Management Server および運用管理エージェントが稼働しているかどうかを確認するための待ち時間を、0 ~ 2147483 の整数（単位：秒）で指定します。

指定した待ち時間を経過しても応答がない場合、コマンドは異常終了します。

0 を指定した場合は無限に待ちます。なお、0 より小さい値を指定した場合は 0 を、2147483 より大きい値を指定した場合は 2147483 が仮定されます。省略時のデフォルトは 0 です。

このオプションは、check サブコマンドを指定した場合だけに有効です。

<サブコマンド> <引数>

サブコマンドと引数を指定します。サブコマンドと引数に指定できる値は、Management Server のバージョンに依存します。

サブコマンドでは、指定する引数によって、ターゲットに論理サーバまたはホストが指定できます。

次のサブコマンドが指定できます。

- list
- get
- set
- start
- tracelevel
- stop
- reload
- hold
- release
- collect
- change
- check
- dump
- add
- delete

なお、サブコマンドの詳細は、「7.3 mngsvrutil コマンドのサブコマンドの詳細」を参照してください。

戻り値

- 0 :
正常終了しました。
- 1 :
実行時エラーが発生しました。
- 2 :
シンタックスエラーが発生しました。

出力形式

CSV 形式

名称、および値はコンマで区切って出力します。
-h オプションでヘッダの出力を抑止できます。

SNMP 連携形式

値は半角スペース (0x20) で区切って出力します。また、文字列はダブルクォーテーション (0x22) で囲んで出力します。

なお、文字列にダブルクォーテーション (0x22) とバックスラッシュ「¥」(0x5c) が含まれていた場合、エスケープコード (バックスラッシュ「¥」(0x5c)) を付加します。

(例)

編集前文字列: " abc " def¥ g

編集後文字列: " ¥" abc¥ " def¥¥ g "

整数値は 32bit 整数型で表します。また、浮動小数点数は小数点以下を切り捨てて 32bit 整数型で表します。

ヘッダは出力されません。

設定ファイル

クライアント側定義ファイル (.mngsvrutilrc)

mngsvrutil コマンドのオプションのデフォルト値を設定できます。

詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編(サーバ定義)」を参照してください。

mngsvrutilrc のパラメタである mngsvrutil.target_kind ,

mngsvrutil.target_name , mngsvrutil.target_server_name プロパティと "-k" , "-t" オプションとの組み合わせの時に有効となる値を次の表に示します。

7. Management Server で使用するコマンド

項番	target_kind	target_name	target_server_name	-k	-t	指定値の解釈
1	logicalServer/ -	-	-	logicalServer/ -	-	サーバ名が指定されていません。
2	logicalServer/ -	A	B/ -	logicalServer/ -	-	“ A ” という名称のサーバが指定されました。
3	logicalServer/ -	-	B	logicalServer/ -	-	“ B ” という名称のサーバが指定されました。
4	logicalServer/ -	A/ -	B/ -	logicalServer/ -	C	“ C ” という名称のサーバが指定されました。
5	host	-	-	-	-	ホスト名が指定されていません。
6	host/ -	-	-	host	-	ホスト名が指定されていません。
7	host	-	B	-	-	ホスト名が指定されていません。
8	host/ -	-	B	host	-	ホスト名が指定されていません。
9	host	A	B/ -	-	-	“ A ” という名称のホストが指定されました。
10	host/ -	A/ -	B/ -	host	C	“ C ” という名称のホストが指定されました。

(凡例)

- : 値を指定しません。

クライアント側共通定義ファイル (mngsvrutilcl.properties)

mngsvrutil コマンドのオプションの共通のデフォルト値を設定できます。

詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス定義編 (サーバ定義)」を参照してください。

サーバ側定義ファイル (mngsvrutil.properties)

mngsvrutil コマンドの実行環境を変更できます。

詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス定義編 (サーバ定義)」を参照してください。

入力例・出力例

入力例 1

運用管理ドメイン内の論理サーバとステータス情報の一覧の取得

```
$ mngsvrutil -m 123.45.67.89:28880 -u admin1 -p pw1 list status
```

出力例 1

```
logicalServerName,status,type,startingOrder,parent
```

```

smplSvrName1,working,NamingService,10,-
smplSvrName2,starting,SFOServer,15,-
smplSvrName3,stopped,J2EEServer,20,J2EEServerCluster1
smplSvrName4,starting,J2EEServer,30,J2EEServerCluster1
smplSvrName5,stopping,WebServer,40,WebServerCluster1
smplSvrName6,abnormal ended,J2EEServer,50,-
smplSvrName7,recovering,J2EEServer,60,-
smplSvrName8,communication
error,J2EEServer,70,J2EEServerCluster2
...

```

入力例 2**運用管理ドメイン名称の取得**

```
$ mngsvrutil -m localhost -u admin1 -p pw1 get domainName
```

出力例 2

```
smplDomainName
```

入力例 3**J2EEアプリケーションについての情報の取得**

```
$ mngsvrutil -m 123.45.67.89 -u admin2 -t myServer get j2eeApps
```

出力例 3

```

index,fullyQualifiedname,name,deployedTime
1,serverName1:smplApp1,smplApp1,1234567890
2,serverName2:smplApp2,smplApp2,9876543210
...
7,serverName7:smplApp7_TEST,smplApp7,7656543210

```

入力例 4**Stateless Session Bean のホームインタフェースについての情報の取得**

```
$ mngsvrutil -m 123.45.67.89 -u admin2 -t myServer get
statelessBeanHome
```

出力例 4

```

index,fullyQualifiedName,name,responseMaxTime,responseMinTime,re
sponseAveTime,statisticsStartedTime,samplingTime
1,smpQfn1,smplBeanName1,300,100,123.45,67339847,600
2,smpQfn2,smplBeanName2,17,52,34.251,122343838,10
...

```

入力例 5**ホスト内のすべての J2EE アプリケーションに対応するスケジュールキューの閉塞**

```
$ mngsvrutil -m 123.45.67.89 -u admin2 -t hostA -k host hold queues
in:-1
```

入力例 6**ホスト内のすべての J2EE アプリケーションに対応するスケジュールキューの閉塞解除**

```
$ mngsvrutil -m 123.45.67.89 -u admin2 -t hostA -k host release
queues
```

7. Management Server で使用するコマンド

入力例 7

運用管理ドメイン内のすべてのパフォーマンストレーサが収集した性能解析トレースの一括収集

```
$ mngsvrutil -m 123.45.67.89 -u admin2 collect allPrfTraces
```

SNMP 連携時の設定手順 (UNIX の場合)

UNIX の場合の SNMP 連携時の設定手順を次に示します。

1. JP1/Cm2/ESA のインストール

2. ~root/.mngsvrutilrc ファイルの作成, 編集

- /opt/Cosminexus/manager/config/templates/.mngsvrutilrc ファイルを ~root にコピーします。

```
# cp /opt/Cosminexus/manager/config/templates/.mngsvrutilrc  
~root/
```

- ~root/.mngsvrutilrc ファイルのファイルパーミッションを 600 に変更します。

```
# chmod 600 ~root/.mngsvrutilrc
```

- ~root/.mngsvrutilrc ファイルを編集します。

適切なエディタを使用して、次の項目を設定します。各項目の詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」を参照してください。

```
mngsvrutil.connect.host  
mngsvrutil.connect.userid  
mngsvrutil.connect.password  
mngsvrutil.output.format=snmp  
mngsvrutil.target_name  
mngsvrutil.target_kind
```

3. JP1/Cm2/ESA の環境設定

Cosminexus の拡張 MIB オブジェクト定義ファイル (/opt/Cosminexus/manager/config/hitachi-cosmiMng-MIB¹⁾,²⁾) を使用して、SNMP サブエージェントを再構成します。SNMP サブエージェントの再構成の方法については、マニュアル「JP1/Cm2/Extensible SNMP Agent」を参照してください。

注 1 実際は、hitachi-cosmiMng-MIB のあとに _V-R (バージョン) が付加されます (例: hitachi-cosmiMng-MIB_2-0)。

注 2 Cosminexus Manager 06-00 以降で提供している拡張 MIB オブジェクト定義ファイルを使用する場合は、hitachi-cosmiMng-MIB_V-R_forESA を使用してください。

拡張 MIB オブジェクト定義ファイルは最新のバージョンを使用してください。

注意事項

コマンドを入力する際、オプションおよびサブコマンドを次のように指定します。

- オプションのサブコマンドおよび引数は、必ず最後尾に指定してください。

- オプションを重複して指定した場合は、最後に指定したものを有効とします。
- ある特定の操作に対して無意味なオプションを指定した場合（例えば、set サブコマンドで `-h` オプションを指定）、そのオプションは無視されます。
- オプションの指定が省略された場合、クライアント側定義ファイル（`.mngsvrutilrc`）にデフォルト値が設定されているときは、そのデフォルト値を有効にします。
- すべてのオプションを省略した場合、Usage (Simple Usage) を表示します。また、`-m`、`-u`、および `-p` オプション以外の引数をすべて省略した場合、詳細な Usage (detailed usage) を表示します。

list または get で取得した値が異常値（統計を取得していないときの統計値など）の場合、値は -1 を 0 として表示します。

サブコマンドで取得した情報は標準出力に表示します。

Usage およびエラー情報は標準エラー出力に表示します。

SNMP 連携時の設定で、SNMP Manager が参照する拡張 MIB オブジェクト定義ファイルのバージョンが、SNMP Agent が参照する拡張 MIB オブジェクト定義ファイルのバージョンよりも新しい場合は、期待する結果が得られない場合があります。このため、同じバージョンに合わせるか、SNMP Agent が参照する拡張 MIB オブジェクト定義ファイルのバージョンを、SNMP Manager が参照する拡張 MIB オブジェクト定義ファイルのバージョンより新しくする必要があります。

mstrexport (Management Server 管理ファイルの退避)

形式

mstrexport <ファイル名> [<退避対象定義ファイル名>]

機能

運用管理サーバの実行系と待機系を 1:1 にして系切り替えをする場合、実行系の運用管理サーバの Management Server 管理ファイルを収集して ZIP 形式ファイルに出力し、退避します。

引数

オプションの指定がない場合、コマンドの使用方法が表示されます。

<ファイル名>

Management Server 管理ファイルを退避する時に出力する ZIP ファイル名を指定します。絶対パス、相対パスで指定できます。拡張子を「.zip (大文字、または小文字)」にしなかった場合は、「.zip」を付加します。

<退避対象定義ファイル名>

7. Management Server で使用するコマンド

Management Server 管理ファイル用退避対象定義ファイル名を絶対パス、または相対パスで指定します。

Management Server 管理ファイル用退避対象定義ファイルは、デフォルトで退避対象となっているファイルのほかに収集したいファイルがある場合に指定します。詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」を参照してください。

入力例

入力例 1

カレントディレクトリに Management Server 管理ファイルを収集した mstruct.zip を出力する場合

```
mstrexport mstruct.zip
```

入力例 2

カレントディレクトリ以外のディレクトリに、デフォルトで収集されるファイルに加えて filelist.txt に記述されたファイルを収集した mstruct.zip を出力する場合

- Windows の場合

```
mstrexport c:/tmp/mstruct.zip "c:/Documents and Settings/  
MyUser/filelist.txt"
```

- UNIX の場合

```
mstrexport /tmp/mstruct.zip "/usr/filelist.txt"
```

注意事項

mstrexport コマンドで退避したファイルのアクセス権は、デフォルトのアクセス権となります。そのため、一般ユーザの読み取りを許可しないなど、アクセス制限を設ける必要のあるファイルは退避対象に含めないでください。

mstrimport (Management Server 管理ファイルの回復)

形式

```
mstrimport <ファイル名>
```

機能

運用管理サーバの実行系と待機系を 1:1 にして系切り替えをする場合に、mstrexport コマンドで退避した実行系の運用管理サーバの管理ファイル (ZIP 形式) を待機系の運用管理サーバで回復します。

回復に成功したファイルのパス名は、システムプロパティを含んだ形式で標準出力に出力されます。

このコマンドは Management Server が停止している場合に実行できます。

引数

オプションの指定がない場合、コマンドの使用方法が表示されます。

< ファイル名 >

Management Server 管理ファイルの退避コマンドで退避した Management Server 管理ファイル名 (ZIP 形式) を指定します。絶対パス、または相対パスで指定できます。

Management Server 管理ファイルの退避コマンドで退避していないファイルを指定した場合や、ファイルが展開できない場合はエラーになります。エラーが発生した場合は、処理が中断されます。

入力例

入力例 1

カレントディレクトリの mstruct.zip の内容を展開する場合

```
mstrimport mstruct.zip
```

入力例 2

カレントディレクトリ以外にある mstruct.zip の内容を展開する場合

- Windows の場合

```
mstrimport c:/tmp/mstruct.zip
```

- UNIX の場合

```
mstrimport /tmp/mstruct.zip
```

注意事項

mstrimport コマンドは既存のファイルを上書きします。そのため、コマンド実行前にファイルのバックアップを取得することをお勧めします。バックアップを取得するには、mstrexport コマンドを使用します。待機系で mstrimport コマンドを実行したときに上書き対象となるファイルをすべて退避するには、待機系で実行する mstrexport コマンドの引数に、実行系で実行した mstrexport コマンドに指定した退避対象定義ファイルを指定します。なお、実行系での mstrexport コマンド実行時に退避対象定義ファイルを指定している場合は、待機系での mstrexport コマンド実行時にも退避対象定義ファイルを指定してください。このとき指定する退避対象定義ファイルは、実行系のバックアップ時に指定した退避対象定義ファイルと同じファイルを指定します。

また、UNIX の場合、mstrimport コマンドで回復したファイルには、実行権限が付きません。必要に応じて実行権限を付与してください。

7.3 mngsvrutil コマンドのサブコマンドの詳細

ここでは、mngsvrutil コマンドの引数として指定できる、サブコマンドについて説明します。

(1) list

引数に指定された値のリストを表示します。次の引数を指定できます。

表 7-2 サブコマンド list に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
status	運用管理ドメイン内の論理サーバ名とステータス情報を列挙します。	-	-
appStatus	"-t" オプションで指定された論理サーバにインポートされている J2EE アプリケーション名とステータス情報を列挙します。		-
resStatus	"-t" オプションで指定された論理サーバにインポートされている J2EE リソース名とステータス情報を列挙します。		-
resJavaBeansStatus	"-t" オプションで指定された論理サーバにインポートされている JavaBeans リソース名とステータス情報を列挙します。		-
queues	"-t" オプションで指定された CTM, またはホストに属するキュー名とステータス情報を列挙します。		

(凡例)

- : 指定できます。
- : 指定できません。

list では、次の 5 種類の情報を表示できます。

- 引数に "status" を指定して、運用管理ドメイン内の論理サーバとそのステータス情報を表示します。
出力情報のヘッダ部の type は論理サーバの種類 (Web サーバ, J2EE サーバ, ネーミングサービス, スマートエージェント, パフォーマンストレーサ, CTM ドメインマネージャ, CTM, および SFO サーバ), startingOrder は論理サーバの起動順序, parent は論理サーバが属するクラスタ名を表します。起動順序を設定していない場合, startingOrder にはデフォルト値が表示され, 起動順序情報を取得できない場合は「-」が表示されます。デフォルト値については, マニュアル「Cosminexus アプリケーションサーバ 機能解説 運用 / 監視 / 連携編」を参照してください。クラスタに属さない単体の論理サーバの場合, Parent には「-」が表示されます。
status の種類は, working (稼働中), stopped (停止), starting (起動中),

stopping (停止中), abnormal ended (異常停止), recovering (回復中), communication error (通信障害), auto stopping (自動停止中), auto restarting (自動再起動中), graceful stopping (計画停止中), または force stopping (強制停止中) のどれかです。論理サーバのステータスの詳細については、マニュアル「Cosminexus アプリケーションサーバ 運用管理ポータル操作ガイド」を参照してください。このサブコマンドでは、クラスタに属している論理サーバと単体の論理サーバだけを表示します。

- 引数に "appStatus" を指定して、J2EE サーバにインポートされている J2EE アプリケーションの情報の一覧を表示します。
"-t" オプションで指定された J2EE サーバにインポートされている J2EE アプリケーションの情報を一覧表示します。
J2EE サーバクラスタが指定された場合、メンバに処理を展開します。"-t" オプションに J2EE サーバ (J2EE サーバクラスタのメンバも含む) または J2EE サーバクラスタ以外の論理サーバが指定された場合はエラーになります。
出力情報のヘッダ部の logicalServerName には、"-t" オプションで指定した J2EE サーバ名が表示されます。
出力情報のヘッダ部の appName には、J2EE サーバにインポートされているアプリケーションの表示名が表示されます。なお、J2EE サーバが起動していない場合は、空文字 ("") が表示されます。J2EE サーバが起動していて、インポートされたアプリケーションがない場合は、レコードを出力しません。
status の種類は、working (稼働中), stopped (停止), または unknown (J2EE サーバが起動していない、もしくは J2EE サーバで管理されていない) のどれかです。
- 引数に "resStatus" を指定して、J2EE サーバ、またはバッチサーバにインポートされている J2EE リソースの情報の一覧を表示します。
"-t" オプションで指定された J2EE サーバ、またはバッチサーバにインポートされている J2EE リソースの情報を一覧表示します。
J2EE サーバクラスタが指定された場合、メンバに処理を展開します。"-t" オプションに J2EE サーバ (J2EE サーバクラスタのメンバも含む)、バッチサーバまたは J2EE サーバクラスタ以外の論理サーバが指定された場合はエラーになります。
出力情報のヘッダ部の logicalServerName には、"-t" オプションで指定した J2EE サーバ名、またはバッチサーバ名が表示されます。
出力情報のヘッダ部の resourceName には、J2EE サーバにインポートされているリソースアダプタの表示名が表示されます。なお、J2EE サーバが起動していない場合は、空文字 ("") が表示されます。
status の種類は、working (稼働中), stopped (停止), または unknown (J2EE サーバが起動していない、もしくは J2EE サーバで管理されていない) のどれかです。
- 引数に "resJavaBeansStatus" を指定して、J2EE サーバ、またはバッチサーバにインポートされている JavaBeans リソースの情報の一覧を表示します。
"-t" オプションで指定された J2EE サーバ、またはバッチサーバにインポートされている JavaBeans リソースの情報を一覧表示します。
J2EE サーバクラスタが指定された場合、メンバに処理を展開します。"-t" オプション

7. Management Server で使用するコマンド

に J2EE サーバ (J2EE サーバクラスタのメンバも含む)、バッチサーバまたは J2EE サーバクラスタ以外の論理サーバが指定された場合はエラーになります。

出力情報のヘッダ部の `logicalServerName` には、"-t" オプションで指定した J2EE サーバ名、またはバッチサーバ名が表示されます。

出力情報のヘッダ部の `JavaBeansResourceName` には、J2EE サーバにインポートされている JavaBeans リソースの表示名が表示されます。J2EE サーバが起動していない場合は、空文字 ("") が表示されます。

`status` の種類は、`working` (稼働中)、`stopped` (停止)、または `unknown` (J2EE サーバが起動していない、もしくは J2EE サーバで管理されていない) のどれかです。

- 引数に "queues" を指定して、CTM のキューの情報を表示します。

"-t" オプションで指定された CTM、またはホストに所属するキューの情報を一覧表示します。

ヘッダ部の `logicalServerName` には CTM 名称が表示されます。`queueName` には キュー名称が表示されます。

`status` の種類は、`scheduling` (スケジューリングが可能な状態)、`preparing` (キュー準備中)、`stopping` (キュー終了処理中)、`closing[in]` (キュー入り口閉塞中)、または `closing[out]` (キュー出口閉塞中) のどれかです。

注意事項

V7 互換モードで動作している場合、引数に "appStatus" を指定すると登録日時とインポート日時が表示されます。また、引数に "resStatus" または "resJavaBeansStatus" を指定するとインポート日時が表示されます。なお、V7 互換モード以外で登録日時およびインポート日時を表示する場合はヘッダ形式互換オプションに "true" を設定してください。ヘッダ部が表示され、項目には「0」が表示されます。

(2) get

"-t" オプションで指定された論理サーバまたはホストについて、引数に指定された値を取得します。表 7-3 に示す引数を指定できます。なお、`domainName` 以外の引数を、`<metric category>` といいます。

"-t" オプションに論理サーバを指定した場合、`<metric category>` に該当する論理サーバ (J2EE サーバ、SFO サーバ、CTM) 以外の論理サーバ (Web サーバ、Naming Service、スマートエージェントなど) が指定された場合は、エラーになります。

ある一定間隔で統計情報を監視する場合、サンプリング時間をその一定間隔以上に設定すると、定期的な統計情報を取得できます。

なお、J2EE アプリケーションがテストモードで開始されている場合、出力情報には、`<アプリケーション名>_TEST` の形式で出力されます。

サブコマンド `get` に指定できる引数を次に示します。サブコマンド `get` に指定できる各引数の詳細については、マニュアル「Cosminexus アプリケーションサーバリファレン

ス 定義編 (サーバ定義)」を参照してください。

表 7-3 サブコマンド get に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
domainName	運用管理ドメイン名を取得します。	-	-
j2eeContainer	J2EE コンテナについての情報を取得します。		
webContainer	Web コンテナに関する情報を取得します。		
jvm	JavaVM についての情報を取得します。		
transaction	Transaction についての情報を取得します。		
httpServer	管理用サーバ (簡易 Web サーバ) についての情報を取得します。 なお、この情報は内部で使用する管理用サーバの情報のため、取得しないことを推奨します。		
httpSvrConn	Web サーバと Web コンテナの接続についての情報を取得します。		
inprocHttpSvr	インプロセス HTTP サーバについての情報を取得します。		
dataSource	データソース (JDBC) についての情報を取得します。		
dataSrcFull	データソース (OTS 1.2 対応の JTA) についての情報を取得します。		
resAdapter	リソースアダプタについての情報を取得します。		
j2eeApps	J2EE アプリケーションについての情報を取得します。		
ejbApps	EJB アプリケーション (EJB-JARs) についての情報を取得します。		
statelessBean	Stateless Session Bean についての情報を取得します。		
statelessBeanHome	Stateless Session Bean のリモートホームインタフェースについての情報を取得します。		
statelessBeanHomeMethod	Stateless Session Bean のリモートホームインタフェースのメソッドに関する情報を取得します。		
statelessBeanLocalHome	Stateless Session Bean のローカルホームインタフェースについての情報を取得します。		
statelessBeanLocalHomeMethod	Stateless Session Bean のローカルホームインタフェースのメソッドに関する情報を取得します。		
statelessBeanRemote	Stateless Session Bean のリモートコンポーネントインタフェースについての情報を取得します。		
statelessBeanRemoteMethod	Stateless Session Bean のリモートコンポーネントインタフェースのメソッドに関する情報を取得します。		
statelessBeanLocalComponent	Stateless Session Bean のローカルコンポーネントインタフェースについての情報を取得します。		

7. Management Server で使用するコマンド

引数	説明	ターゲット指定	
		論理サーバ	ホスト
statelessBeanLocalComponentMethod	Stateless Session Bean のローカルコンポーネントインタフェースのメソッドに関する情報を取得します。		
statefulBean	Stateful Session Bean についての情報を取得します。		
statefulBeanHome	Stateful Session Bean のリモートホームインタフェースについての情報を取得します。		
statefulBeanHomeMethod	Stateful Session Bean のリモートホームインタフェースのメソッドに関する情報を取得します。		
statefulBeanLocalHome	Stateful Session Bean のローカルホームインタフェースについての情報を取得します。		
statefulBeanLocalHomeMethod	Stateful Session Bean のローカルホームインタフェースのメソッドに関する情報を取得します。		
statefulBeanRemote	Stateful Session Bean のリモートコンポーネントインタフェースについての情報を取得します。		
statefulBeanRemoteMethod	Stateful Session Bean のリモートコンポーネントインタフェースのメソッドに関する情報を取得します。		
statefulBeanLocalComponent	Stateful Session Bean のローカルコンポーネントインタフェースについての情報を取得します。		
statefulBeanLocalComponentMethod	Stateful Session Bean のローカルコンポーネントインタフェースのメソッドに関する情報を取得します。		
entityBean	Entity Bean についての情報を取得します。		
entityBeanHome	Entity Bean のリモートホームインタフェースについての情報を取得します。		
entityBeanLocalHome	Entity Bean のローカルホームインタフェースについての情報を取得します。		
entityBeanRemote	Entity Bean のリモートコンポーネントインタフェースについての情報を取得します。		
entityBeanLocalComponent	Entity Bean のローカルコンポーネントインタフェースについての情報を取得します。		
msgDrivenBean	Message-driven Bean についての情報を取得します。		
webApps	Web アプリケーションについての情報を取得します。		
servlets	サーブレットについての情報を取得します。		
requestUrl	要求された各 URL についての情報を取得します。		
queues	CTM のスケジュール (キューの状態) についての情報を取得します。		
queueApps	CTM のキューに接続された J2EE アプリケーションについての情報を取得します。		

(凡例)

- : 指定できます。
- : 指定できません。

(3) set

引数を二つ指定して、一つ目の引数に指定された変数に二つ目で指定された値を設定します。

値を設定したあと、対象の論理 J2EE サーバ、論理 SFO サーバ、または J2EE アプリケーションが再起動または再デプロイされた場合、値を再設定してください。"-t" オプションに論理 J2EE サーバ、論理 SFO サーバ、または論理パフォーマンストレーサ以外の論理サーバが指定された場合は、エラーとなります。

サブコマンド set に指定できる引数を次に示します。

表 7-4 サブコマンド set に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
jvmSamplingTime <n>	JavaVM についてのサンプリング時間を <n> 秒に設定します。 ¹		-
webContainerSamplingTime <n>	Web コンテナについてのサンプリング時間を <n> 秒に設定します。 ¹		-
tranSamplingTime <n>	トランザクションについてのサンプリング時間を <n> 秒に設定します。 ^{1, 2}		-
httpSvrSamplingTime <n>	管理用サーバ (簡易 Web サーバ) についてのサンプリング時間を <n> 秒に設定します。 ¹ なお、管理用サーバ (簡易 Web サーバ) の情報は内部で使用する管理用サーバの情報のため、取得しないことを推奨します。		-
httpSvrConnSamplingTime <n>	HTTP サーバコネクタについてのサンプリング時間を <n> 秒に設定します。 ¹		-
inprocHttpSvrSamplingTime <n>	インプロセス HTTP サーバについてのサンプリング時間を <n> 秒に設定します。 ¹		-
dataSrcFullSamplingTime <n>	データソース (OTS 1.2 対応の JTA) についてのサンプリング時間を <n> 秒に設定します。 ^{1, 3}		-
resAdapterSamplingTime <n>	リソースアダプタについてのサンプリング時間を <n> 秒に設定します。 ^{1, 3}		-
ejbSamplingTime <n>	EJB についてのサンプリング時間を <n> 秒に設定します。 ^{1, 3}		-
webAppSamplingTime <n>	Web アプリケーションについてのサンプリング時間を <n> 秒に設定します。 ^{1, 3}		-

7. Management Server で使用するコマンド

引数	説明	ターゲット指定	
		論理サーバ	ホスト
servletSamplingTime <n>	サーブレットについてのサンプリング時間を <n> 秒に設定します。 ^{1, 3}		-
reqUrlSamplingTime <n>	設定以降に要求される各 URL のサンプリング時間を <n> 秒にします。 ^{1, 3}		-
allPrfTraceLevel STANDARD DETAIL <tracelevel>	運用管理ドメイン内のすべてのパフォーマンストレーサについて PRF トレース取得レベルを設定します。 ⁴	-	-
prfTraceLevel STANDARD DETAIL <tracelevel>	指定されたパフォーマンストレーサ、または指定されたホスト内のパフォーマンストレーサの PRF トレース取得レベルを設定します。 ⁴		

(凡例)

- : 指定できます。
- : 指定できません。

注 1

-2147483647 n 3600 (秒) の整数値を指定します。この範囲以外の値が指定された場合はエラーとなります。なお、n 0 の場合は統計値取得スイッチを OFF にします。また、0<n<10 の場合は n=10 とします。指定値による動作を次に示します。

0 以下の値が指定された場合は、統計を停止します。

1 以上 9 以下の値が指定された場合は、サンプリング時間をデフォルト (10 秒) に設定して、統計を開始します。

10 以上の値が指定された場合は、サンプリング時間を指定された値に設定して、統計を開始します。

注 2

最初のトランザクション開始後に設定する必要があります。

注 3

この変数を設定後に J2EE アプリケーションをデプロイした場合、新たに設定する必要があります。

注 4

パフォーマンストレーサへの操作については、稼働中のパフォーマンストレーサが対象となります。

STANDARD

PRF トレース取得レベルを「標準」に設定します。

DETAIL

PRF トレース取得レベルを「詳細」に設定します。

(4) tracelevel

機能レイヤごとに異なる PRF トレース取得レベル指定します。4 バイト (8 けた) 以下の 16 進数をコンマ (,) 区切りで最大 16 個指定できます。指定値の詳細については、「4.2 性能解析トレースで使用するコマンドの詳細」の「cprfststart (PRF デーモンの開始)」の -PrfTraceLevel の説明、または「cprflevel (PRF トレース取得レベルの表示と変更)」の -PrfTraceLevel の説明を参照してください。

(5) start

指定した論理サーバ、J2EE アプリケーション、J2EE リソースを起動します。

サブコマンド start に指定できる引数を次に示します。

表 7-5 サブコマンド start に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
allServers	運用管理ドメインに定義された論理サーバを定義された起動順序で起動します。	-	-
server	"-t" オプションで指定された論理サーバを起動します。		
app <displayName>	"-t" オプションで指定された論理サーバにインポートされている <displayName> の J2EE アプリケーションを開始します。		-
resAdapter <displayName>	"-t" オプションで指定された論理サーバにインポートされている <displayName> の J2EE リソースアダプタを開始します。		-
resJavaBeans	"-t" オプションで指定された論理サーバにインポートされている <displayName> の JavaBeans リソースを開始します。		-

(凡例)

- : 指定できます。
- : 指定できません。

注

<displayName> は、J2EE アプリケーション、J2EE リソースまたは JavaBeans リソースの表示名 (displayName) を示します。

start では、次の 2 種類の操作を実行できます。

- 引数に "allServers" を指定して、指定された論理サーバを起動します。
運用管理ドメインに定義された論理サーバを、定義された起動順序で起動します。
引数に server が指定された場合、"-t" オプションで指定された論理サーバ (スマートエージェント、ネーミングサービス、J2EE サーバ、J2EE サーバクラスタ、CTM、

7. Management Server で使用するコマンド

CTM ドメインマネージャ、パフォーマンスストレサ、Web サーバ、Web サーバクラスタ、SFO サーバ、およびユーザサーバ)を起動します。

- 引数には start サブコマンドに対する引数を二つ指定して、指定された J2EE アプリケーション / J2EE リソースアダプタ / JavaBeans リソースを開始します。
一つ目の引数に "app"/"resAdapter"/"resJavaBeans" が指定された場合、"-t" オプションで指定された J2EE サーバにインポートされている、二つ目の引数で指定された表示名 (displayname) の J2EE アプリケーション / リソースアダプタ / JavaBeans リソースを開始します。J2EE サーバクラスタの場合は、メンバとなる J2EE サーバに操作を展開します。
"-t" オプションに J2EE サーバ (J2EE サーバクラスタのメンバも含む) または J2EE サーバクラスタ以外の論理サーバが指定された場合はエラーになります。また、"-t" オプションにホスト名は指定できません。
なお、J2EE アプリケーションは、テストモードで開始できません。

(6) stop

指定した論理サーバ、J2EE アプリケーションまたは J2EE リソースを停止します。

サブコマンド stop に指定できる引数を次に示します。

表 7-6 サブコマンド stop に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
allServers	運用管理ドメインに定義された論理サーバを定義された起動順序の逆順で停止します。	-	-
server [graceful[:<waitTime>]]	"-t" オプションで指定された論理サーバを停止します。 "graceful" が指定された場合、計画停止します。		
app <displayName> [force]	"-t" オプションで指定された論理サーバで開始している <displayName> の J2EE アプリケーションを停止します。 "force" が指定された場合、強制停止します。		-
resAdapter <displayName>	"-t" オプションで指定された論理サーバで開始している <displayName> の J2EE リソースを停止します。		-
resJavaBeans	"-t" オプションで指定された論理サーバで開始している <displayName> の JavaBeans リソースを停止します。		-

(凡例)

- : 指定できます。
- : 指定できません。

注

<displayName> は、J2EE アプリケーションまたは J2EE リソースまたは JavaBeans リソースの表示名 (displayName) を示します。

stop では、次の 2 種類の操作を実行できます。

- 引数に "allServers" を指定して、指定された論理サーバを停止します。
運用管理ドメインに定義された論理サーバを、定義された起動順序の逆順で停止します。引数に "server" が指定された場合、"-t" オプションで指定された論理サーバ (スマートエージェント、ネーミングサービス、J2EE サーバ、J2EE サーバクラスタ、CTM、CTM ドメインマネージャ、パフォーマンスストレサ、Web サーバ、Web サーバクラスタ、SFO サーバ、およびユーザサーバ) を停止します。
一つ目の引数に "server" が指定された場合、二つ目の引数に "graceful" を指定できます。"graceful" が指定された場合、"-t" オプションで指定された論理サーバが Web サーバまたは Web サーバクラスタの場合、論理サーバを計画停止します。計画停止とは、現在受け付けているすべてのリクエストの処理終了を待って論理サーバを停止させることです。"-t" オプションで指定された論理サーバが Web サーバおよび Web サーバクラスタ以外の場合、または "-t" オプションにホスト名を指定した場合は、引数 "graceful" は無視されます。また、引数 "graceful" では ":<waitTime>" を指定して停止までの待ち時間を指定できます。"<waitTime>" には、60 ~ 1800 (単位：秒) の値を指定します。指定した待ち時間の間にリクエストが終了しなかった場合、論理サーバを強制停止します。":<waitTime>" を省略した場合は、運用管理ポータルの「論理サーバの起動 / 停止」で設定した「停止監視時間」が設定されます。
- 引数に stop サブコマンドに対する引数を二つ指定して、指定された J2EE アプリケーション / J2EE リソースアダプタ / JavaBeans リソースを停止します。
一つ目の引数に "app"/"resAdapter"/"resJavaBeans" が指定された場合、"-t" オプションで指定された J2EE サーバで、二つ目の引数で指定された表示名 (displayname) の開始している J2EE アプリケーション / リソースアダプタ / JavaBeans リソースを停止します。J2EE サーバクラスタの場合はメンバとなる J2EE サーバに操作を展開します。
"-t" オプションに J2EE サーバ (J2EE サーバクラスタのメンバも含む) または J2EE サーバクラスタ以外の論理サーバが指定された場合は、エラーになります。また、"-t" オプションにホスト名は指定できません。
一つ目の引数に "app" が指定された場合、三つ目の引数に "force" を指定できます。引数 "force" が指定された場合、J2EE アプリケーションがリクエスト処理の実行中であっても、その終了を待たないで論理サーバを強制停止します。
テストモードで開始されている J2EE アプリケーションは停止できません。

(7) reload

引数に指定されたコンポーネントの設定ファイルを再読み込みします。

サブコマンド reload に指定できる引数を次に示します。

表 7-7 サブコマンド reload に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
userAdmin	統合ユーザ管理で使用する設定ファイルを再読み込みします。なお、"-t" オプションで指定された論理サーバで統合ユーザ管理を使用していない場合はエラーとなります。 また、サブコマンド "reload" に "userAdmin" を指定した場合、"-t" オプションに、論理 J2EE サーバ以外の論理サーバを指定するとエラーとなります。		-
env	論理 J2EE サーバ、または論理 SFO サーバで、接続先ホストの設定ファイルを再読み込みします。なお、サブコマンド "reload" に "env" を指定した場合、"-t" オプションに、論理 J2EE サーバ、および論理 SFO サーバ以外の論理サーバを指定するとエラーとなります。		

(凡例)

- : 指定できます。
- : 指定できません。

注

サブコマンド "reload" に "env" を指定する場合には、注意事項があります。注意事項については、マニュアル「Cosminexus アプリケーションサーバ 運用管理ポータル操作ガイド」の、論理 SFO サーバの設定読み込みに関する説明、および論理 J2EE サーバの設定読み込みに関する説明を参照してください。

(8) hold

引数に指定されたサービスを閉塞します。

サブコマンド hold に指定できる引数を次に示します。

表 7-8 サブコマンド hold に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
queue <queueName> { in[:<waitTime>] out force[:<waitTime>]}	"-t" オプションで指定された CTM に属する <queueName> で指定された名称のキューを閉塞します。		-
queues { in[:<waitTime>] out force[:<waitTime>]}	"-t" オプションで指定された CTM、またはホストに属するキューを閉塞します。		
allQueues { in[:<waitTime>] out force[:<waitTime>]}	運用管理ドメイン内の、すべての CTM のキューを閉塞します。	-	-

(凡例)

- : 指定できます。
- : 指定できません。

hold では、次の 3 種類の方法でキューを閉塞できます。

- 引数に "queue" を指定して、特定の名称のキューを閉塞します。

"-t" オプションで指定された CTM に属する <queueName> の名称のキューを閉塞します。"-t" オプションおよび <queueName> は必ず指定してください。なお、"-t" オプションにホスト名は指定できません。

キューの閉塞条件として、"in" (キューの入り口を閉塞)、"out" (キューの出口を閉塞) または "force" (キューの強制閉塞) のどれかを指定してください。"in" 指定の場合、キューイング中のリクエスト終了の待ち時間を指定できます。待ち時間は、"in:<waitTime>" の形式で、次の値を指定します。

0 :

キューの入り口を閉塞し、キューイング中のすべてのリクエストが終了するまで待ちます。

1 ~ 2147483647 :

キューの入り口を閉塞し、キューイング中のリクエストの終了を指定秒数まで待ちます。

-1 :

キューの入り口を閉塞し、キューイング中のリクエストはすべてエラーリターンします。

"in" だけ指定した場合は、"in:-1" が設定されます。

"out" を指定し、キューの出口を閉塞した場合、閉塞中もリクエストは受け付けられませんが、リクエストは J2EE アプリケーションに振り分けられません。オンライン中に J2EE アプリケーションの入れ替えをする場合には、"out" を指定してキューを閉塞してください。

"force" を指定した場合は、"in:-1" を指定したときと同様にキューの入り口を閉塞し、キューイング中のリクエストはすべてエラーリターンしますが、"in:-1" を指定したときよりも高速に閉塞できます。また、"force:<waitTime>" の形式で待ち時間を指定できます。指定する値は、"in" 指定の場合と同じです。

- 引数に "queues" を指定して、複数のキューを閉塞します。

"-t" オプションで指定された CTM、またはホストに属するすべてのキューを閉塞します。"-t" オプションは必ず指定してください。"in", "out", "force" の指定は "queue" の場合と同じです。

- 引数に "allQueues" を指定して、すべてのキューを閉塞します。

運用管理ドメインのすべてのキューを閉塞します。"in", "out", "force" の指定は "queue" の場合と同じです。

(9) release

引数に指定されたサービスの閉塞を解除します。

サブコマンド release に指定できる引数を次に示します。

表 7-9 サブコマンド release に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
queue <queueName>	"-t" オプションで指定された CTM に属する <queueName> で指定された名称のキューの閉塞を解除します。		-
queues	"-t" オプションで指定された CTM、またはホストに属するキューの閉塞を解除します。		
allQueues	運用管理ドメイン内の、すべての CTM のキューの閉塞を解除します。	-	-

(凡例)

- : 指定できます。
- : 指定できません。

release では、次の 3 種類の方法でキューの閉塞を解除できます。

- 引数に "queue" を指定して、特定の名称のキューの閉塞を解除します。
"-t" オプションで指定された CTM に属する名称 <queueName> のキューの閉塞を解除します。"-t" オプションおよび <queueName> は必ず指定します。なお、"-t" オプションにホスト名は指定できません。
- 引数に "queues" を指定して、複数のキューの閉塞を解除します。
"-t" オプションで指定された CTM、またはホストに属するキューの閉塞を解除します。"-t" オプションは必ず指定してください。
- 引数に "allQueues" を指定して、すべてのキューの閉塞を解除します。
運用管理ドメインのすべてのキューの閉塞を解除します。

(10) collect

引数に指定されたファイルなどを収集します。"-t" オプションに、パフォーマンストレーサ、CTM 以外の論理サーバが指定された場合は、エラーになります。

サブコマンド collect に指定できる引数を次に示します。

表 7-10 サブコマンド collect に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
allPrfTraces	運用管理ドメイン内のすべてのパフォーマンストレーサについて性能解析トレーサを収集します。	-	-
prfTrace	指定されたパフォーマンストレーサ, または指定されたホスト内のパフォーマンストレーサについて性能解析トレーサを収集します。		
allCtmStatistics	運用管理ドメイン内のすべての CTM について稼働統計情報を収集します。	-	-
ctmStatistics	指定された CTM, または指定されたホスト内の CTM について稼働統計情報を収集します。		
snapshot [n]	指定されたホスト内の snapshot ログを収集します。	-	

(凡例)

- : 指定できます。
- : 指定できません。

注

[n] は, 次のどれかを指定します。

- 1: 一次送付資料となる snapshot ログを収集します。
 - 2: 二次送付資料となる snapshot ログおよび性能解析トレーサファイルを収集します。
- 省略: すべてを対象に snapshot ログおよび性能解析トレーサファイルを収集します。

collect では, 次の 3 種類の操作を実行できます。

- 引数に "allPrfTraces" または "prfTrace" を指定して, 稼働中のパフォーマンストレーサの性能解析トレーサを収集します。
引数に "allPrfTraces" が指定された場合, 運用管理ドメイン内のすべてのパフォーマンストレーサについて性能解析トレーサを収集します。
引数に "prfTrace" が指定された場合, "-t" オプションで指定されたパフォーマンストレーサ, または "-t" オプションで指定されたホスト内のパフォーマンストレーサについて性能解析トレーサを収集します。
- 引数に "allCtmStatistics" または "ctmStatistics" を指定して, 稼働中の CTM の稼働統計情報を収集します。
引数に "allCtmStatistics" が指定された場合, 運用管理ドメイン内のすべての CTM について稼働統計情報を収集します。
引数に "ctmStatistics" が指定された場合, "-t" オプションで指定された CTM, または "-t" オプションで指定されたホスト内の CTM について稼働統計情報を収集します。
CTM 起動ホストで CSV 形式で出力された CTM 稼働統計情報の収集ファイルは, ZIP 形式でアーカイブされ, <ホスト名><論理サーバ名><日時>.zip ファイルが作成されます。さらに, Management Server にファイル転送されたあと, <Manager

7. Management Server で使用するコマンド

ログ出力ディレクトリ >¥ctm¥ (Windows の場合), または <Manager ログ出力ディレクトリ >/ctm/ (UNIX の場合) に格納されます。ドメイン指定またはホスト指定の場合は複数ファイルを ZIP 形式の一つのファイルにして、論理サーバ指定の場合はファイル名を変更して格納します。

- ドメイン指定の場合 : <ドメイン名><日時>.zip
- ホスト指定の場合 : <ホスト名><日時>.zip
- 論理サーバ指定の場合 : <論理サーバ名><日時>.zip
- 引数に "snapshot" を指定して、snapshot ログを収集します。
引数に "snapshot" が指定された場合、"-t" オプションで指定されたホスト内の snapshot ログを収集します。同時に、インストール情報および OS 状態情報が取得されます。なお、"-t" オプションに、論理サーバが指定された場合はエラーとなります。収集が成功すると、ZIP 形式で作成された次のファイル名を表示します。
 - 一次送付資料 : snapshot-<ホスト名>-log-<yyyyMMddHHmmss>.zip
 - 二次送付資料 : snapshot-<ホスト名>-log-<yyyyMMddHHmmss>.2.zip

(11)change

引数に指定した同時実行スレッド数の設定項目の値を変更します。"-t" オプションで、J2EE サーバを指定する必要があります。

サブコマンド change に指定できる引数を次に示します。

表 7-11 サブコマンド change に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
webAppThreadCtrl <webApplicationContextRoot> <maxThreads>,<exclusiveThreads>,<queueSize>	"-t" オプションで指定された J2EE サーバ内で指定した Web アプリケーションの同時実行スレッド数を動的に制御します。 この引数は省略できません。		-

(凡例)

- : 指定できます。
- : 指定できません。

引数に "webAppThreadCtrl" が指定された場合、"-t" オプションで指定された J2EE サーバに属する Web アプリケーションの同時実行スレッド数を動的に制御します。

引数 "webApplicationContextRoot" は Web アプリケーション (J2EE アプリケーションの War) のコンテキストルートおよび同時実行スレッド数の各値を設定します。Web アプリケーションのコンテキストルートの値は、サブコマンド get webApps で取得できません。コンテキストルートが設定されていない場合は、"/" (スラッシュ) を指定してくだ

さい。

同時実行スレッド数の各値は、最大スレッド数、占有スレッド数、Web アプリケーション単位の実行待ちキューサイズの順にコンマ(,)で区切って指定します。

引数 "maxThreads" (最大スレッド数), "exclusiveThreads" (占有スレッド数), および "queueSize" (Web アプリケーション単位の実行待ちキューサイズ) は、設定を変えない値は省略できますが、すべてを省略することはできません。

(例 占有スレッド数だけを変更したい場合: ,20,)

同時スレッド数の各値は、次の範囲で指定します。

- maxThreads (最大スレッド数) :
1 ~ 1024 の範囲で指定します。
Web コンテナ単位の最大同時実行スレッド数以下の値を指定します。
- exclusiveThreads (占有スレッド数) :
0 ~ 1024 の範囲で指定します。
- queueSize (Web アプリケーション単位の実行待ちキューサイズ):
0 ~ 2147483647 の範囲で指定します。

注 最大スレッド数 >= 占有スレッド数

なお、動的変更した同時実行スレッド数の値は、J2EE アプリケーション実行中だけ有効です。

(12) check

Management Server および運用管理エージェントが稼働しているかどうかを確認します。

サブコマンド check に指定できる引数を次に示します。

表 7-12 サブコマンド check に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
mngsvr	Management Server が稼働しているかどうかを確認します。 "-t" オプションでターゲットを指定しても無視されます。	-	-

7. Management Server で使用するコマンド

引数	説明	ターゲット指定	
		論理サーバ	ホスト
adminAgent	"-t" オプションで指定されたホスト上で運用管理エージェントが稼働しているかどうかを確認します。 "-k" オプションで "host" 以外を指定した場合、"-t" オプションでホスト名以外を指定した場合、または "-t" オプションを指定しなかった場合は、エラーとなります。	-	

(凡例)

- : 指定できます。
- : 指定できません。

サブコマンド check の戻り値を次に示します。

0 :

稼働しています。

1 :

コマンド実行時エラーです。

2 :

コマンドシンタックスエラーです。

103 :

稼働していません。

注 -w オプションで指定した稼働確認待ち時間が経過しても Management Server からの応答が返ってこなかった場合、サブコマンド check に指定した引数によって、戻り値は次のように異なります。

- mngsvr を指定した場合、終了値 103 が返されます。
- adminAgent を指定した場合、終了値 1 が返されます。

(13) dump

稼働中の論理サーバの javacore を出力します。

サブコマンド dump に指定できる引数を次に示します。

表 7-13 サブコマンド dump に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
server [addition]	"-t" オプションで指定された論理サーバの javacore (稼働時間情報) を出力します。指定できる論理サーバは、J2EE サーバ (J2EE クラスタのメンバも含む)、J2EE サーバクラスタ、SFO サーバ、CTM、またはネーミングサービスだけです。 addition を指定した場合、"-t" オプションで指定された論理サーバの core またはメモリダンプ情報を取得します。		-

(凡例)

- : 指定できます。
- : 指定できません。

注

server addition が指定された場合、指定された論理サーバの稼働監視方法はプロセスの存在確認だけに切り替わります。

javacore 出力のために実行されるコマンドと javacore ファイルの出力先について、OS ごとに説明します。

(a) Windows の場合

Windows の場合に javacore 出力のために実行されるコマンドと javacore ファイルの出力先を論理サーバの種別ごとに示します。

論理サーバ	addition 指定	実行されるコマンド	出力情報	javacore ファイルの出力先ディレクトリ
J2EE サーバ (クラスタも含む), SFO サーバ	あり	cjdumpsv, userdump ¹	javacore, メモリダンプ	<J2EE サーバ作業ディレクトリ>¥ejb¥<実サーバ名>¥
	なし	cjdumpsv	javacore	<J2EE サーバ作業ディレクトリ>¥ejb¥<実サーバ名>¥
CTM	あり, なし ²	hdumpns	javacore ³	<Cosminexus インストールディレクトリ>¥TPB¥log¥
ネーミングサービス	あり, なし ²	hdumpns	javacore	<Cosminexus インストールディレクトリ>¥TPB¥log¥

注 1

userdump コマンドがインストールされている必要があります。

注 2

addition 指定は無視され、稼働監視方法の変更はありません。

7. Management Server で使用するコマンド

注 3

論理サーバに CTM を指定したときに出力される javacore は CTM デーモンの情報ではなく、CTM 用に起動されているネーミングサービスの情報です。

(b) UNIX の場合

UNIX の場合に javacore 出力のために実行されるコマンドと javacore ファイルの出力先を論理サーバの種別ごとに示します。

論理サーバ	addition 指定	実行されるコマンド	出力情報	javacore ファイルの出力先ディレクトリ
J2EE サーバ(クラスタも含む), SFO サーバ	あり	javacore -f -p ¹	javacore , core	<J2EE サーバ作業ディレクトリ >/ejb/ <実サーバ名 >/
	なし	cjdumpsv	javacore	<J2EE サーバ作業ディレクトリ >/ejb/ <実サーバ名 >/
CTM	あり, なし ²	kill -3	javacore ₃	/opt/Cosminexus/TPB/logj/
ネーミングサービス	あり, なし ²	kill -3	javacore	/opt/Cosminexus/TPB/logj/

注 1

gdb のパッケージがインストールされている必要があります。

注 2

addition 指定は無視され、稼働監視方法の変更はありません。

注 3

論理サーバに CTM を指定したときに出力される javacore は CTM デーモンの情報ではなく、CTM 用に起動されているネーミングサービスの情報です。

(14) add

論理ユーザサーバを追加します。

サブコマンド add に指定できる引数を次に示します。

表 7-14 サブコマンド add に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
userserver <userServiceDefinitionFile> e>	<userServiceDefinitionFile> に指定した論理ユーザサーバ定義ファイルの内容に従って論理ユーザサーバを追加します。論理ユーザサーバの定義内容を変更する場合は、論理ユーザサーバをいったん削除したあとに、再度論理ユーザサーバを追加してください。	-	-

(凡例)

- : 指定できません。

複数の論理ユーザサーバを指定した場合、定義順に追加処理をします。途中でエラーが発生した場合も追加処理をします。この場合の戻り値は、1 (実行時エラー) となります。エラー条件を次に示します。

- 既存の論理サーバ名が指定された場合
- 定義されていないホスト名が指定された場合

(15)delete

論理ユーザサーバを削除します。

サブコマンド delete に指定できる引数を次に示します。

表 7-15 サブコマンド delete に指定できる引数

引数	説明	ターゲット指定	
		論理サーバ	ホスト
userserver <logicalServerName>	<logicalServerName> に指定した名称の論理ユーザサーバを削除します。	-	-

(凡例)

- : 指定できません。

8

Smart Composer 機能で使用するコマンド

この章では、Smart Composer 機能で使用するコマンドの入力形式、機能などについて説明します。

8.1 Smart Composer 機能で使用するコマンドの一覧

8.2 コマンドを同時に複数実行した場合の排他制御

8.3 Smart Composer 機能で使用するコマンドの詳細

8.1 Smart Composer 機能で使用するコマンドの一覧

ここでは、Smart Composer 機能で使用するコマンドの一覧について説明します。

8.1.1 J2EE アプリケーションを実行するシステムで使用できるコマンドの一覧

J2EE アプリケーションを実行するシステムで使用できるコマンドの一覧を次の表に示します。

表 8-1 Smart Composer 機能で使用するコマンドの一覧 (J2EE アプリケーションを実行するシステムの場合)

分類	コマンド名称	概要
環境設定	cmx_admin_passwd	Management Server の管理ユーザアカウントを設定します。または、設定済みの管理ユーザアカウントを変更、解除します。
Web システムの構築	cmx_build_model	Management Server 上に Web システムの情報モデルを生成します。
	cmx_build_system	Management Server 上の Web システムの情報モデルから Web システムを構築します。
	cmx_change_model	Management Server 上の Web システムの情報モデルを変更します。
	cmx_delete_system	構築した Web システムや Web システム中のサービスユニットを削除します。
	cmx_resume_lb	負荷分散機の休止状態を解除して、設定処理を再開します。
	cmx_scaleout_host	Management Server 上の、ホスト単位管理モデルの Web システムの情報モデルを変更します。
	cmx_test_lb	負荷分散機との接続に問題がないか、接続テストをします。
	cmx_trans_param	抽象パラメタを実パラメタに展開します。
サービスユニットの操作	cmx_list_status	サービスユニットの稼働状況のステータスや運用環境の構築状況を表示します。
	cmx_start_target	単一のサービスユニット、または Web システム中の全サービスユニットを開始します。
	cmx_stop_target	単一のサービスユニット、または Web システム中の全サービスユニットを停止します。

分類	コマンド名称	概要
構成情報の参照	cmx_list_model	Management Server 上に構築した Web システムの情報モデルの内容を CSV 形式で出力します。
構築済みのシステムの設定内容の出力	cmx_export_model	構築済みのシステムの設定内容を、簡易構築定義ファイルの形式で出力します。

注

Smart Composer 機能で使用するコマンドの実行状況は、アプリケーションサーバのログ（コマンド保守ログ）として出力されます。アプリケーションサーバのログ（コマンド保守ログ）については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」のトラブルシューティングに関する説明を参照してください。

8.1.2 バッチアプリケーションを実行するシステムで使用できるコマンドの一覧

バッチアプリケーションを実行するシステムの場合、使用できるコマンドが一部異なります。バッチアプリケーションを実行するシステムで使用できるコマンドの一覧を次の表に示します。

表 8-2 Smart Composer 機能で使用するコマンドの一覧（バッチアプリケーションを実行するシステムの場合）

分類	コマンド名称	概要
環境設定	cmx_admin_passwd	Management Server の管理ユーザアカウントを設定します。または、設定済みの管理ユーザアカウントを変更、解除します。
Web システムの構築	cmx_build_model	Management Server 上に Web システムの情報モデルを生成します。
	cmx_build_system	Management Server 上の Web システムの情報モデルから Web システムを構築します。
	cmx_change_model	Management Server 上の Web システムの情報モデルを変更します。
	cmx_delete_system	構築した Web システムや Web システム中のサービスユニットを削除します。
	cmx_scaleout_host	Management Server 上の、ホスト単位管理モデルの Web システムの情報モデルを変更します。
サービスユニットの操作	cmx_list_status	サービスユニットの稼働状況のステータスや運用環境の構築状況を表示します。
	cmx_start_target	単一のサービスユニット、または Web システム中の全サービスユニットを開始します。
	cmx_stop_target	単一のサービスユニット、または Web システム中の全サービスユニットを停止します。

8. Smart Composer 機能で使用するコマンド

分類	コマンド名称	概要
構成情報の参照	cmx_list_model	Management Server 上に構築した Web システムの情報モデルの内容を CSV 形式で出力します。
構築済みのシステムの設定内容の出力	cmx_export_model	構築済みのシステムの設定内容を、簡易構築定義ファイルの形式で出力します。

注

Smart Composer 機能で使用するコマンドの実行状況は、アプリケーションサーバのログ（コマンド保守ログ）として出力されます。アプリケーションサーバのログ（コマンド保守ログ）については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」のトラブルシューティングに関する説明を参照してください。

バッチアプリケーションを実行するシステムの場合は、次の点に留意してください。

- バッチアプリケーションを実行するシステムでは、サービスユニットとして j2ee-tier または ctm-tier を使用します。それ以外のティアの説明は、バッチアプリケーションを実行するシステムでは該当しません。
- Smart Composer 機能のコマンドでは、バッチサーバを J2EE サーバとして扱います。このため、この章の説明中の「J2EE サーバ」は、「バッチサーバ」に読み替えてください。

8.1.3 旧バージョンとの互換用のコマンドの一覧

ここでは、Smart Composer 機能で使用する旧バージョンとの互換用のコマンドの一覧について説明します。

(1) J2EE アプリケーションを実行するシステムで使用できる互換用コマンドの一覧

J2EE アプリケーションを実行するシステムで使用できる互換用コマンドの一覧を次に示します。

表 8-3 Smart Composer 機能で使用する互換用コマンドの一覧（J2EE アプリケーションを実行するシステムの場合）

分類	コマンド名称	概要
サービスユニットの操作	cmx_add_serverref ¹	論理ユーザサーバをサービスユニットに追加します。
	cmx_delete_serverref ²	論理ユーザサーバをサービスユニットから削除します。

分類	コマンド名称	概要
リソースの操作 ³	cmx_define_resource	Management Server に登録したリソースアダプタまたは JavaBeans リソースを Web システムに関連づけます。
	cmx_deploy_resource	Web システムに関連づけたリソースアダプタまたは JavaBeans リソースをデプロイします。
	cmx_register_resource	リソースアダプタまたは JavaBeans リソースを Management Server に登録します。
	cmx_start_resource	デプロイ済みのリソースアダプタまたは JavaBeans リソースを開始します。
	cmx_stop_resource	開始済みのリソースアダプタまたは JavaBeans リソースを停止します。
	cmx_undefine_resource	リソースアダプタまたは JavaBeans リソースと Web システムの関連づけを解除します。
	cmx_undeploy_resource	デプロイ済みのリソースアダプタまたは JavaBeans リソースをアンデプロイします。
	cmx_unregister_resource	リソースアダプタまたは JavaBeans リソースを Management Server から削除します。
アプリケーションの操作 ³	cmx_define_application	Management Server に登録した J2EE アプリケーションを Web システムに関連づけます。
	cmx_deploy_application	Web システムに関連づけた J2EE アプリケーションをデプロイします。
	cmx_register_application	J2EE アプリケーションを Management Server に登録します。
	cmx_start_application	デプロイ済みの J2EE アプリケーションを開始します。
	cmx_stop_application	開始済みの J2EE アプリケーションを停止します。
	cmx_undefine_application	J2EE アプリケーションと Web システムの関連づけを解除します。
	cmx_undeploy_application	デプロイ済みの J2EE アプリケーションをアンデプロイします。
	cmx_unregister_application	J2EE アプリケーションを Management Server から削除します。

注 1

論理サーバ参照定義ファイルを使用する場合だけに使用します。07-50 以降は、論理ユーザサーバは、簡易構築定義ファイルで設定することをお勧めします。

注 2

論理ユーザサーバだけを削除したい場合に使用します。07-50 以降は、cmx_delete_system コマンドで削除することをお勧めします。

注 3

8. Smart Composer 機能で使用するコマンド

リソースおよびアプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。なお、リソースおよびアプリケーションの操作に関するコマンドは、統合ネーミングスケジューラサーバ用の Web システムでは使用できません。統合ネーミングスケジューラサーバ用の Web システムでこれらのコマンドを実行した場合は、エラーになります。

(2) バッチアプリケーションを実行するシステムで使用できる互換用コマンドの一覧

バッチアプリケーションを実行するシステムで使用できる互換用コマンドの一覧を次に示します。

表 8-4 Smart Composer 機能で使用する互換用コマンドの一覧 (バッチアプリケーションを実行するシステムの場合)

分類	コマンド名称	概要
リソースの操作	cmx_define_resource	Management Server に登録したリソースアダプタまたは JavaBeans リソースを Web システムに関連づけます。
	cmx_deploy_resource	Web システムに関連づけたリソースアダプタまたは JavaBeans リソースをデプロイします。
	cmx_register_resource	リソースアダプタまたは JavaBeans リソースを Management Server に登録します。
	cmx_start_resource	デプロイ済みのリソースアダプタまたは JavaBeans リソースを開始します。
	cmx_stop_resource	開始済みのリソースアダプタまたは JavaBeans リソースを停止します。
	cmx_undefine_resource	リソースアダプタまたは JavaBeans リソースと Web システムの関連づけを解除します。
	cmx_undeploy_resource	デプロイ済みのリソースアダプタまたは JavaBeans リソースをアンデプロイします。
	cmx_unregister_resource	リソースアダプタまたは JavaBeans リソースを Management Server から削除します。

注

リソースの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。なお、リソースの操作に関するコマンドは統合ネーミングスケジューラサーバ用の Web システムでは使用できません。統合ネーミングスケジューラサーバ用の Web システムでこれらのコマンドを実行した場合は、エラーになります。

8.2 コマンドを同時に複数実行した場合の排他制御

Smart Composer 機能のコマンドを同時に複数実行した場合、コマンドによって、あとから実行したコマンドが排他制御されるものと、排他制御されないものがあります。

Smart Composer 機能のコマンドについて、排他制御の対象となるかどうかを次の表に示します。

表 8-5 排他制御の対象 / 対象外のコマンド

コマンドの種類	コマンド名	排他制御の対象
Web システム全体に対するコマンド	cmx_build_model	
	cmx_change_model	
	cmx_scaleout_host	
個別の Web システムに対するコマンド	cmx_admin_passwd	x
	cmx_build_system	
	cmx_delete_system	
	cmx_export_model	x
	cmx_list_model	x
	cmx_list_status	x
	cmx_resume_lb	
	cmx_start_target	
	cmx_stop_target	
	cmx_test_lb	
	cmx_trans_param	x

(凡例)

：排他制御の対象です。これらのコマンドは、「8.3 Smart Composer 機能で使用するコマンドの詳細」の共通引数にある、-wait/-nowait オプションが指定できるコマンドです。

x：排他制御の対象外です。実行中のコマンドがある場合でも実行できます。

次に、Smart Composer 機能の旧バージョンとの互換用のコマンドについて、排他制御の対象となるかどうかを示します。

表 8-6 排他制御の対象 / 対象外の互換用のコマンド

コマンドの種類	コマンド名	排他制御の対象
Web システム全体に対するコマンド	cmx_add_serverref	
	cmx_delete_serverref	
	cmx_register_application	

8. Smart Composer 機能で使用するコマンド

コマンドの種類	コマンド名	排他制御の対象
	cmx_register_resource	
	cmx_unregister_application	
	cmx_unregister_resource	
個別の Web システムに対するコマンド	cmx_define_application	
	cmx_define_resource	
	cmx_deploy_application	
	cmx_deploy_resource	
	cmx_start_application	
	cmx_start_resource	
	cmx_stop_application	
	cmx_stop_resource	
	cmx_undefine_application	
	cmx_undefine_resource	
	cmx_undeploy_application	
	cmx_undeploy_resource	

(凡例)

: 排他制御の対象です。これらのコマンドは、「8.3 Smart Composer 機能で使用するコマンドの詳細」の共通引数にある、-wait/-nowait オプションが指定できるコマンドです。

排他制御の対象となるコマンドでは、コマンドの共通引数の -wait/-nowait オプションの指定によって、複数のコマンドを同時に実行したときの動作が変わります。また、コマンドの種類や、どの Web システムを対象としてコマンドを実行するかの組み合わせで、排他制御される場合とされない場合があります。

次の表に、複数のコマンドを同時に実行したときの排他制御のルールを示します。なお、表中の「Web システム全体に対するコマンド」および「個別の Web システムに対するコマンド」に分類されるコマンドの種類については、表 8-5 および表 8-6 を参照してください。

表 8-7 排他制御のルール

あとから実行するコマンド			実行中のコマンド	
コマンドの種類	対象となるシステム	共通引数の指定	コマンドの種類	
			Web システム全体に対するコマンド	個別の Web システムに対するコマンド
Web システム全体に対するコマンド	-	-wait		
		-nowait	x	x

あとから実行するコマンド			実行中のコマンド	
コマンドの種類	対象となるシステム	共通引数の指定	コマンドの種類	
			Web システム全体に対するコマンド	個別の Web システムに対するコマンド
個別の Web システムに対するコマンド	同じ Web システム	-wait		
		-nowait	x	x
	異なる Web システム	-wait		
		-nowait	x	

(凡例)

- x : あとから実行するコマンドは排他エラーが返ります。
- : あとから実行するコマンドは実行中のコマンドが終了するまで待ちます。
- : あとから実行するコマンドを実行します。
- : 該当しません。

8.3 Smart Composer 機能で使用するコマンドの詳細

Smart Composer 機能で使用するコマンドの入力形式、機能などを次に示します。

コマンドの格納先

Smart Composer 機能で使用するコマンドは、次のディレクトリに格納されています。

- Windows の場合
`<Cosminexus のインストールディレクトリ >%manager%bin%`
- UNIX の場合
`/opt/Cosminexus/manager/bin/`

共通引数

各コマンドには、次の引数を共通引数として指定します。共通引数は、`.cmxrc` (クライアント設定プロパティファイル) または `cmxclient.properties` (クライアント共通設定ファイル) で、指定を省略したときのデフォルト値を設定できます。`.cmxrc` ファイルおよび `cmxclient.properties` ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」の `cmxrc` (クライアント設定プロパティファイル) および `cmxclient.properties` (クライアント共通設定プロパティファイル) の説明を参照してください。

形式

```

コマンド名称 [-m <ホスト名>[:<ポート番号>]]
                [-u <管理ユーザID>] [-p <管理ユーザパスワード>]
                [-wait | -nowait] [-verbose | -noverbose]

```

説明

`-m <ホスト名>[:<ポート番号>]`

Management Server のホスト名とポート番号を指定します。

- <ホスト名>
 ドット記法での IP アドレス指定 (`xxx.xxx.xxx.xxx`)、または名前解決できるホスト名で指定します。なお、`xxx` には 0 ~ 255 の整数を指定します。
- <ポート番号>
 Management Server の設定に従います。デフォルト値は 28080 です。

`-u <管理ユーザ ID>`

Management Server のセットアップで設定した、Management Server の管理ユーザ ID を指定します。Management Server のセットアップについては、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」を参照してください。管理ユーザアカウントの省略機能が有効の場合、管理ユーザ ID を指定しても無効となります。管理ユーザアカウントの省略機能については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運

用ガイド」の管理ユーザアカウントの設定を省略する場合に関する説明を参照してください。

-p <管理ユーザパスワード>

Management Server のセットアップで設定した、Management Server の管理ユーザパスワードを指定します。Management Server のセットアップについては、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」を参照してください。管理ユーザアカウントの省略機能が有効の場合、管理ユーザパスワードを指定しても無効となります。管理ユーザアカウントの省略機能については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の管理ユーザアカウントの設定を省略する場合に関する説明を参照してください。

-wait | -nowait

コマンド実行時に、すでに実行しているコマンドとの排他が必要な場合、処理できるようになるまで待つか (-wait)、待たずにエラーリターンするか (-nowait) のどちらかを指定します。詳細は、「8.2 コマンドを同時に複数実行した場合の排他制御」を参照してください。

-verbose | -noverbose

コマンドの実行結果の詳細を出力するか (-verbose)、出力しないか (-noverbose) のどちらかを指定します。

共通引数は、コマンドによって指定が有効になる場合と、無効になる場合があります。コマンドに無効な引数を指定した場合、その引数は無視されます。共通引数の指定について、コマンドごとに次の表に示します。

表 8-8 共通引数の指定の可否

コマンド名称	共通引数				
	-m	-u	-p	-wait -nowait	-verbose -noverbose
cmx_admin_passwd				x	x
cmx_build_model					x
cmx_build_system					
cmx_change_model					x
cmx_delete_system					
cmx_export_model				x	x
cmx_list_model				x	x
cmx_list_status				x	x
cmx_resume_lb					
cmx_scaleout_host					
cmx_start_target					
cmx_stop_target					

8. Smart Composer 機能で使用するコマンド

コマンド名称	共通引数				
	-m	-u	-p	-wait -nowait	-verbose -noverbose
cmx_test_lb					
cmx_trans_param	x	x	x	x	x

(凡例)

：有効です。

x：無効です。指定した場合、無視されます。

注

互換用のコマンドはすべての共通引数の指定が有効です。互換用のコマンドについては、「8.1.3 旧バージョンとの互換用のコマンドの一覧」を参照してください。

終了コード

Smart Composer 機能で使用するコマンドは、次の範囲で戻り値を返します。

表 8-9 Smart Composer 機能で使用するコマンドの戻り値

戻り値	意味
0	コマンドの処理が正常終了しました。
1 ~ 63	コマンドの処理で軽微な障害が発生しました。
64 ~ 127	コマンドの処理が異常終了しました。

cmx_add_serverref (論理ユーザサーバの追加)(互換用)

形式

cmx_add_serverref [共通引数] -f <論理サーバ参照定義ファイル>

機能

cmx_add_serverref コマンドを使用すると、論理サーバ参照定義ファイルの内容に従って、Web システム中のサービスユニットにユーザ定義の論理サーバ (論理ユーザサーバ) を追加します。

論理サーバ参照定義ファイルで複数の論理ユーザサーバが指定されている場合は、定義順にサービスユニットへ追加します。追加した論理ユーザサーバに対するエラーチェックは、処理開始時に実行されます。

引数

-f < 論理サーバ参照定義ファイル >

論理サーバ参照定義ファイルのパスを指定します。

入力例

```
cmx_add_serverref -f UserServerRef.xml
```

戻り値

- 0 :
正常終了しました。
- 1 :
警告終了しました。
- 64 :
異常終了しました。

注意事項

指定した論理ユーザサーバが、同じサービスユニット、またはほかのサービスユニットに追加済みの場合はエラーになります。

論理ユーザサーバの追加先に指定した Web システム、およびサービスユニットが存在しない場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。論理サーバ参照定義ファイルを使用して、構築済みのシステムに論理ユーザサーバを追加する場合だけに使用します。07-50 以降は、簡易構築定義ファイルで論理ユーザサーバを設定することを勧めます。

cmx_admin_passwd (Management Server の管理ユーザアカウントの設定)

形式

```
cmx_admin_passwd [共通引数] [-np <新管理ユーザパスワード>] [-nu <新管理ユーザID>]
```

機能

Management Server へログインするための管理ユーザアカウントを設定します。新管理ユーザアカウントを指定している場合は、設定済みの管理ユーザアカウントを新管理ユーザアカウントに変更します。また、-np オプションに空文字 ("") を指定している場合は、設定済みの管理ユーザパスワードを解除します。

引数

- np <新管理ユーザパスワード>
新しい Management Server の管理ユーザパスワードを指定します。
- nu <新管理ユーザ ID>

8. Smart Composer 機能で使用するコマンド

新しい Management Server の管理ユーザ ID を指定します。

入力例

- 新規に管理ユーザアカウントを設定する場合

```
cmx_admin_passwd -m localhost -u admin01 -p p1574809
```

- 設定済みの管理ユーザアカウントを変更する場合

```
cmx_admin_passwd -m localhost -u admin01 -p p1574809 -np passwd123  
-nu admin02
```

- 管理ユーザパスワードを解除する場合

```
cmx_admin_passwd -m localhost -u admin01 -p p1574809 -np ""
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

管理ユーザパスワードが設定されていない状態で、`-np` オプションを指定したコマンドを入力すると、共通引数の `-p` オプションの指定は無視されて、新管理ユーザパスワードが設定されます。また、管理ユーザ ID が設定されていない状態で、`-nu` オプションを指定したコマンドを入力すると、共通引数の `-u` オプションの指定は無視されて、新管理ユーザ ID が設定されます。管理ユーザアカウントの省略機能を有効にしている場合、メッセージを表示し、終了コード 0 を出力します。また、管理ユーザアカウントの設定はしません。管理ユーザアカウントの省略機能については、マニュアル「Cosminexus アプリケーションサーバ システム構築・運用ガイド」の管理ユーザアカウントの設定を省略する場合に関する説明を参照してください。

cmx_build_model (Web システムの情報モデルの生成)

形式

```
cmx_build_model [共通引数] -f <簡易構築定義ファイル>
```

機能

`cmx_build_model` コマンドを使用すると、簡易構築定義ファイルの内容に従って、Management Server にホストおよび論理サーバを定義し、Web システムの情報モデル

を構築します。

cmx_build_model コマンドで Management Server 上に論理サーバ名が定義される場合のルールについては、「cmx_build_system」を参照してください。

引数

-f < 簡易構築定義ファイル >

簡易構築定義ファイルのパスを指定します。

入力例

```
cmx_build_model -f MyWebSystemDef.xml
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

構築済みの Web システムを処理対象に指定した場合はエラーになります。

07:50 以降は、cmx_build_system コマンドを使用して、Web システムの情報モデルの生成から、Web システムの構築まで一括で実行することをお勧めします。

cmx_build_system (Web システムの構築)

形式

Web システムの構築

- 簡易構築定義ファイルから Web システムを構築する場合

```
cmx_build_system [ 共通引数 ] -f < 簡易構築定義ファイル >
                    [-lb {keepvs|deletevs}]
```

- 登録済みの情報モデルを基に Web システムを構築する場合

```
cmx_build_system [ 共通引数 ] [-s <Web システム名 >]
                    [-lb {keepvs|deletevs}]
```

- Web システムに関連づいていない論理サーバを含むシステムを構築する場合

```
cmx_build_system [ 共通引数 ] -ns
```

構築済みのシステムのパラメタ変更

- 構築済みのシステムのパラメタを簡易構築定義ファイルで変更する場合

8. Smart Composer 機能で使用するコマンド

```
cmx_build_system [ 共通引数 ] -f < 簡易構築定義ファイル > -change
```

機能

Web システムを構築します。また、構築済みのシステムのパラメタを変更します。

Web システムの構築

登録済みの Web システムの情報モデルに従って、Web システムを構築します。または、簡易構築定義ファイルの内容に従って、Web システムの情報モデルを生成し、その情報モデルに従って Web システムを構築します。

cmx_build_system コマンドは、各ホストの Web サーバと J2EE サーバのセットアップ、および各論理サーバの環境を設定します。また、必要に応じて負荷分散機に仮想サーバや実サーバを作成します。

登録済みの情報モデルを基に Web システムを構築する場合、簡易構築定義ファイルから Web システムを構築する場合、または Web システムに関連づいていない論理サーバを含むシステムを構築する場合で、構築のしかたが異なります。

- 簡易構築定義ファイルから Web システムを構築する場合

指定した簡易構築定義ファイルの内容に従って、Web システムの情報モデルを構築します。さらに、その情報モデルに従って、Web システムを構築します。あらかじめ cmx_build_model でモデルを登録しないで、Web システムの構築ができます。

- 登録済みの情報モデルを基に Web システムを構築する場合

cmx_build_model コマンド、または cmx_change_model コマンドで登録した Web システムの情報モデルに従って、Web システムを構築します。

- Web システムに関連づいていない論理サーバを含むシステムを構築する場合

cmx_build_model コマンド、または cmx_change_model コマンドで登録した Web システムの情報モデルに従って、Web システムに関連づいていない論理サーバを含むシステムを構築します。

構築済みのシステムのパラメタ変更

-change オプションを指定することで、構築済みのシステムのパラメタを、-f オプションで指定した簡易構築定義ファイルに定義されたパラメタに変更します。

- -change オプションを指定して構築済みのシステムのパラメタを変更する場合

-f オプションにパラメタ変更済みの簡易構築定義ファイルを指定して、-change オプションを指定します。-f オプションで指定した簡易構築定義ファイルの内容に従って、Web システムの情報モデルを、パラメタを変更した状態で再生成します。さらに、その情報モデルを基にして、Web システムを再構築します。ただし、構成変更はできません。

Web システムの構築時に、論理サーバ名を簡易構築定義ファイルで定義していない場合、Management Server 上に論理サーバ名が定義されます。なお、登録済みの情報モデルに従って Web システムを構築する場合、論理サーバ名は cmx_build_model コマンド実行時に定義されます。Management Server 上に論理サーバ名が定義されるときルールを次に示します。

論理サーバ名の定義ルール

論理サーバ名は次の形式で定義されます。

形式

cmx_<Webシステム名>_<サービスユニット名>_<種別>_<通番>

<種別> は、論理サーバの種類によって異なります。論理サーバの種類と<種別>との対応を次の表に示します。

表 8-10 論理サーバの種類と<種別>との対応

論理サーバの種類	<種別>
論理 Web サーバ	HTTP
論理 J2EE サーバ	J2EE
論理パフォーマンスストレサ	PRF
論理 SFO サーバ	SFO
論理 CTM ドメインマネージャ	CTMDM
論理 CTM	CTM
論理スマートエージェント	SA
論理ユーザサーバ	USER

<通番> は、基本的に「01」が設定されます。ただし、CTM を使用するシステムのように論理サーバを複数配置する場合は、重複しない番号が割り振られます。

引数

-s <Web システム名>

Web システム名を指定します。

登録済みの情報モデルから、指定した Web システムが構築されます。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-f オプションを指定している場合には指定が無効になります。

-lb

この引数は使用する負荷分散機の種類が BIG-IPv9 の場合だけ有効になります。

負荷分散機の仮想サーバを初期化するかどうかを指定します。JP1/SC/DPM を使用しないで、ホスト単位管理モデルで構築した負荷分散機を使用する Web システムをスケールアウトする場合に指定します。

- keepvs

8. Smart Composer 機能で使用するコマンド

負荷分散機の仮想サーバは初期化しません。複数のホストで負荷分散機の仮想サーバを共有する場合に指定します。

- deletevs

負荷分散機の仮想サーバを初期化します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.lb キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.lb キーを省略している場合に、このオプションを省略したときは deletevs が仮定されます。.cmxrc ファイルおよび cmxclient.properties ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」の .cmxrc (クライアント設定プロパティファイル) および cmxclient.properties (クライアント共通設定プロパティファイル) の説明を参照してください。

-f <簡易構築定義ファイル>

簡易構築定義ファイルのパスを指定します。

簡易構築定義ファイルの内容に従って、Web システムの情報モデルが生成されます。さらに、その情報モデルを基にして Web システムが構築されます。

-change

パラメタを変更する場合に指定します。

-f オプションで指定した簡易構築定義ファイルの内容に従って、Web システムの情報モデルが再生成されます。さらに、その情報モデルを基にして Web システムが再構築されます。

-ns

登録済みの情報モデルから、Web システムに関連していない論理サーバを含むシステムを構築する場合に指定します。

このオプションを指定する場合、cmx_build_model コマンドを使用して、情報モデルを登録しておく必要があります。cmx_build_model コマンドについては、「cmx_build_model」を参照してください。

入力例

- 簡易構築定義ファイルから Web システムを構築する場合
cmx_build_system -f MyWebSystemDef.xml
- 登録済みの情報モデルを基に Web システムを構築する場合
cmx_build_system -s sample-system
- Web システムに関連していない論理サーバを含むシステムを構築する場合
cmx_build_system -ns
- 構築済みのシステムのパラメタを簡易構築定義ファイルで変更する場合
cmx_build_system -f MyWebSystemDef2.xml -change

戻り値

- 0 :
正常終了しました。
- 1 :
警告終了しました。
- 64 :
異常終了しました。

注意事項

-change オプションを指定する場合、次のことに注意してください。

構築済みの Web システムに対してパラメタ変更以外の変更をしようとした場合はエラーになります。

-f オプションで指定した簡易構築定義ファイルのスキーマのバージョンが、構築済みのシステムの構築時に使用した簡易構築定義ファイルのスキーマのバージョンより古いときは、動作が保証されません。

構築済みのシステムの構築時に使用した簡易構築定義ファイルのスキーマのバージョンが 2.1 以前の場合、-f オプションで指定する簡易構築定義ファイルのホストの定義の <host-ref> タグは次のように指定します。

<host-ref> タグに指定する値

構築済みのシステムの構築時に使用した簡易構築定義ファイルのホストの定義の <operational-ip> タグで指定している値

cmx_change_model (Web システムの情報モデルの変更)

形式

cmx_change_model [共通引数] -f <構成変更定義ファイル>

機能

構成変更定義ファイルの内容に従って、Web システムの情報モデルを変更（サービスユニット・ホストの追加またはパラメタの一括変更）します。

- サービスユニット・ホストの追加

Web システムの情報モデルに、構成変更定義ファイルで定義したサービスユニットまたはホストを追加します。サービスユニットに指定するホストには、新規のものも、Management Server 上の Web システムの情報モデルにあるものも指定できます。cmx_change_model コマンドでは、サービスユニットに対応するホストおよび論理サーバを Management Server に定義します。論理サーバ名の設定ルールについては、「cmx_build_system」を参照してください。

8. Smart Composer 機能で使用するコマンド

- パラメタの一括変更
構成変更定義ファイルに指定した論理サーバのコンフィグレーション定義の情報に変更します。

引数

-f <構成変更定義ファイル>

構成変更定義ファイルのパスを指定します。

入力例

```
cmx_change_model -f MyWebSystemDef.xml
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

-f オプションで指定した構成変更定義ファイルのスキーマのバージョンが、構築済みのシステムの構築時に使用した簡易構築定義ファイルのスキーマのバージョンより古いときは、動作が保証されません。

構築済みのシステムの構築時に使用した簡易構築定義ファイルのスキーマのバージョンが 2.1 以前の場合、-f オプションで指定する構成変更定義ファイルのホストの定義の <host-ref> タグは次のように指定します。

<host-ref> タグに指定する値

構築済みのシステムの構築時に使用した簡易構築定義ファイルのホストの定義の <operational-ip> タグで指定している値

cmx_define_application (J2EE アプリケーションと Web システムの関連づけ)(互換用)

形式

- アーカイブ形式の J2EE アプリケーションを関連づける場合

```
cmx_define_application [共通引数] [-s <Webシステム名>] -appname <アプリケーション表示名> -appver <アプリケーションバージョンID>
```

- 展開ディレクトリ形式の J2EE アプリケーションを関連づける場合

```
cmx_define_application [共通引数] [-s <Webシステム名>] -appname <アプリケーション表示名>
```

機能

J2EE アプリケーションを Web システムに関連づけます。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-appname < アプリケーション表示名 >

J2EE アプリケーションの表示名を指定します。cmx_register_application コマンド実行後にメッセージに出力される表示名を使用します。

-appver < アプリケーションバージョン ID >

J2EE アプリケーションのバージョン ID を指定します。

入力例

```
cmx_define_application -s sample-system -appname Appl -appver 1.0
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_define_resource (リソースと Web システムの関連づけ) (互換用)

形式

- リソースアダプタを関連づける場合

```
cmx_define_resource [共通引数] [-s <Webシステム名>] -resname <リソースアダプタ表示名>
```

- JavaBeans リソースを関連づける場合

```
cmx_define_resource [共通引数] [-s <Webシステム名>] -resname <JavaBeansリソース表示名>
```

機能

リソースアダプタまたは JavaBeans リソースを Web システムに関連づけます。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-resname <リソースアダプタ表示名または JavaBeans リソース表示名 >

リソースアダプタまたは JavaBeans リソースの表示名を指定します。

cmx_register_resource コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_define_resource -s sample-system -resname App1
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコマンドは、08-00以降はサーバ管理コマンドを使用することをお勧めします。

cmx_delete_serverref (論理ユーザサーバの削除)(互換用)

形式

```
cmx_delete_serverref [共通引数] [-s <Webシステム名>] -unit <サービスユニット名> -name <論理サーバ名>
```

機能

Web システム中のサービスユニットから論理ユーザサーバを削除します。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-unit < サービスユニット名 >

サービスユニット名を指定します。

-name < 論理サーバ名 >

論理サーバ名を指定します。

入力例

```
cmx_delete_serverref -s sample-system -unit Unit01 -name
cmx_Web02_Unit01_User_01
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

指定した論理ユーザサーバが、同じサービスユニット、またはほかのサービスユニットに追加済みの場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。論理ユーザサーバだけを削除したい場合に使用します。

cmx_delete_system (Web システムの削除)

形式

- Web システムを削除する場合

```
cmx_delete_system [共通引数] [-s <Webシステム名>] [-lb  
{keepvs|deletevs}]
```

- サービスユニットを削除する場合

```
cmx_delete_system [共通引数] [-s <Webシステム名>] -unit <サービスユ  
ニット名>
```

- Web システムに関連づいていない論理サーバ、およびホストをすべて削除する場合

```
cmx_delete_system [共通引数] -ns
```

機能

Web システム、または Web システム中のサービスユニットを削除します。-unit オプションの指定を省略した場合は Web システムを、-unit オプションを指定した場合は Web システム中のサービスユニットを削除します。ホスト単位管理モデルで構築した Web システムの場合に、複数のアプリケーションサーバ (ホスト) で負荷分散機の仮想サーバを共有しているときは、-lb オプションで仮想サーバを削除するかどうかを指定して、Web システムを削除してください。

また、-ns オプションを指定した場合は、Web システムに関連づいていない論理サーバ、およびホストをすべて削除します。

なお、このコマンドは Management Server に登録した Web システムの情報モデル、負荷分散機に設定した情報、および削除対象の Web システムまたはサービスユニットに関連づいている、すべてのホストも削除します。ただし、ほかの Web システムや削除対象外のサービスユニットから参照しているホスト、および Smart Composer 機能以外で論理サーバを追加したホストは削除しません。運用管理ドメイン内に登録されていて、Web システムと関連づいていないホストを削除する場合は、一時的な Web システムやサービスユニットに削除対象とするホストを関連づけて、関連づけた Web システムまたはサービスユニットごと削除する必要があります。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-unit < サービスユニット名 >

サービスユニット名を指定します。

-lb

負荷分散機の仮想サーバを削除するかどうかを指定します。ホスト単位管理モデルで構築した Web システムをスケールアウトする場合に指定します。

- keepvs

負荷分散機の仮想サーバは削除しません。負荷分散機の設定情報を削除します。複数のホストで負荷分散機の仮想サーバを共有している場合に指定します。

- deletevs

負荷分散機の仮想サーバおよび負荷分散機の設定情報を削除します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.lb キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.lb キーを省略している場合に、このオプションを省略したときは deletevs が仮定されます。.cmxrc ファイルおよび cmxclient.properties ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」の .cmxrc（クライアント設定プロパティファイル）および cmxclient.properties（クライアント共通設定プロパティファイル）の説明を参照してください。

-ns

Web システムに関連づいていない論理サーバ、およびホストをすべて削除します。

入力例

```
cmx_delete_system -s sample-system -unit unit1
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

このコマンドで、Web システムやサービスユニットを削除した場合、サービスユニットを構成するホストに Web サーバや J2EE サーバの環境が残ります。

このホストを Smart Composer 機能を使って再利用する場合は、次の方法でサーバの環境をアンセットアップしてください。

- Web サーバの環境のアンセットアップ

Hitachi Web Server のユーティリティを使用してアンセットアップします。

ユーティリティに指定する Web サーバ名は、「HWS_<Web サーバの論理サーバ名>」の形式で指定します。Web サーバの論理サーバ名は cmx_list_model コマンドを使用して参照できます。cmx_delete_system コマンドを実行する前に論理サーバ名を確認してください。

ただし、物理ティアの realservname パラメタで Web サーバの実サーバ名を指定している場合は、ユーティリティに指定するサーバ名は、「HWS_<Web サーバの実サーバ名>」を指定してください。

- J2EE サーバの環境のアンセットアップ

サーバ管理コマンドの cjsetup コマンドを使用してアンセットアップします。

コマンドには、J2EE サーバの論理サーバ名を指定します。J2EE サーバの論理サーバ名は cmx_list_model コマンドを使用して参照できます。cmx_delete_system コマンドを実行する前に論理サーバ名を確認してください。

ただし、物理ティアの realservname パラメタで J2EE サーバの実サーバ名を指定している場合は、コマンドに指定するサーバ名は実サーバ名を指定してください。

それぞれのサーバの環境削除の例を次に示します。

Web サーバの環境削除の例

Windows の場合

```
<Cosminexus インストールディレクトリ>%httpd%admin%bin%hwsserveredit
-delete HWS_cmx_sample-system_unit1_HTTP_01
```

UNIX の場合

```
/opt/hitachi/httpd/admin/bin/hwsserveredit -delete
HWS_cmx_sample-system_unit1_HTTP_01
```

J2EE サーバの環境削除の例

Windows の場合

```
<Cosminexus インストールディレクトリ>%CC%server%bin%cjsetup -d
cmx_sample-system_unit1_J2EE_01
```

UNIX の場合

```
/opt/Cosminexus/CC/server/bin/cjsetup -d cmx_sample-system_unit1_J2EE_01
```

Hitachi Web Server のユーティリティについてはマニュアル「Hitachi Web Server」を参照してください。

このコマンドを実行すると、論理ユーザサーバも同時に削除されます。論理ユーザサーバだけを削除したい場合は、cmx_delete_serverref コマンドを使用してください。

cmx_deploy_application (J2EE アプリケーションのデプロイ)(互換用)

形式

- アーカイブ形式の J2EE アプリケーションを指定する場合

```
cmx_deploy_application [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -appname <アプリケーション表示名> -appver <アプリケーションバージョンID>
```

- 展開ディレクトリ形式の J2EE アプリケーションを指定する場合

```
cmx_deploy_application [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -appname <アプリケーション表示名>
```

- J2EE アプリケーションを指定しない場合

```
cmx_deploy_application [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]
```

機能

J2EE アプリケーションをデプロイします。コマンドに J2EE アプリケーションを指定している場合と、指定していない場合とでデプロイのしかたが異なります。

- アーカイブ形式の J2EE アプリケーションを指定する場合

Web システムに含まれるすべてのサービスユニットに、指定されたアーカイブ形式の J2EE アプリケーションをデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットに、指定されたアーカイブ形式の J2EE アプリケーションをデプロイします。

- 展開ディレクトリ形式の J2EE アプリケーションを指定する場合

Web システムに含まれるすべてのサービスユニットに、指定された展開ディレクトリ形式の J2EE アプリケーションをデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットに、指定された展開ディレクトリ形式の J2EE アプリケーションをデプロイします。

- J2EE アプリケーションを指定しない場合

Web システムに含まれるすべてのサービスユニットに、その Web システムに関連づけられているすべての J2EE アプリケーション (アーカイブ形式および展開ディレクトリ形式) をデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットに、その Web システムに関連づけられているすべての J2EE アプリケーションをデプロイします。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties

8. Smart Composer 機能で使用するコマンド

ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

`-unit <サービスユニット名>`

サービスユニット名を指定します。

`-appname <アプリケーション表示名>`

J2EE アプリケーションの表示名を指定します。`cmx_register_application` コマンド実行後にメッセージに出力される表示名を使用します。

`-appver <アプリケーションバージョン ID>`

J2EE アプリケーションのバージョン ID を指定します。

入力例

```
cmx_deploy_application -s sample-system -unit unit1 -appname Appl  
-appver 1.0
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

アーカイブ形式の J2EE アプリケーションの場合、バージョン ID が異なる、同じ表示名の J2EE アプリケーションを同時にデプロイできません。

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_deploy_resource (リソースのデプロイ)(互換用)

形式

- リソースアダプタを指定してデプロイする場合

```
cmx_deploy_resource [共通引数] [-s <Webシステム名>] [-unit <サービス  
ユニット名>] -resname <リソースアダプタ表示名>
```

- JavaBeans リソースを指定してデプロイする場合

```
cmx_deploy_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -resname <JavaBeansリソース表示名>
```

- リソースアダプタまたは JavaBeans リソースを指定しないでデプロイする場合

```
cmx_deploy_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]
```

機能

リソースアダプタまたは JavaBeans リソースをサービスユニットにデプロイします。コマンドにリソースアダプタまたは JavaBeans リソースを指定している場合と、指定していない場合とでデプロイのしかたが異なります。

- リソースアダプタまたは JavaBeans リソースを指定している場合

Web システムに含まれるすべてのサービスユニットに、指定されたリソースアダプタまたは JavaBeans リソースをデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットに、指定されたリソースアダプタまたは JavaBeans リソースをデプロイします。

- リソースアダプタまたは JavaBeans リソースを指定していない場合

Web システムに含まれるすべてのサービスユニットに、Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースをデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットに、Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースをデプロイします。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-unit <サービスユニット名 >

サービスユニット名を指定します。

-resname <リソースアダプタ表示名または JavaBeans リソース表示名 >

リソースアダプタまたは JavaBeans リソースの表示名を指定します。

cmx_register_resource コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_deploy_resource -s sample-system -unit unit1 -resname App1
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコマンドは、08-00以降はサーバ管理コマンドを使用することをお勧めします。

cmx_export_model (簡易構築定義ファイルの出力)

形式

```
cmx_export_model [共通引数] [-indent <インデントサイズ>] [-myhost <置換するホスト名>] [-encoding <文字エンコーディング名>] -o <出力ファイル>
```

機能

構築されている Web システムの内容に従って、簡易構築定義ファイルを出力します。

引数

-indent <インデントサイズ>

簡易構築定義ファイルのタグのインデントサイズを指定します。下位タグの開始位置を上位のタグから半角スペースで何文字分右にずらすかを 0 ~ 8 の範囲で指定します。

「-indent 4」を指定した場合の出力例

```
<web-system>
  <name>MyWebSystem</name>
  <tier>
    <tier-type>combined-tier</tier-type>
  </tier>
  :
```

(凡例) : 半角スペース 1 文字分

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties

ファイルの `cmx.indent` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.indent` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、2 が設定されます。

-myhost <置換するホスト名>

出力する簡易構築定義ファイルの運用管理サーバマシンのホスト名を指定します。ホスト単位管理モデルのシステムを、別のホストに移行する場合に指定します。このオプションを省略した場合は、`.cmxrc` ファイルまたは `cmxclient.properties` ファイルの `cmx.myhost` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.myhost` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、出力する簡易構築定義ファイルの運用管理サーバマシンのホスト名が設定されません。

なお、移行先の環境で、このオプションの指定が有効になるのは次の場合です。

- ホストの定義の `<host-name>` に、移行先の環境の運用管理サーバマシンの IP アドレスと一致する IP アドレス、または移行先の環境の運用管理サーバマシンの IP アドレスと一致する IP アドレスに変換できるホスト名を指定している。
- ホストの定義の `<host-name>` に `localhost` かループバックアドレスを指定している。

これらの条件を満たさないホスト名を指定した場合、指定は無効になります。指定する値は、移行先の環境の運用管理サーバマシンのホスト名、または `@myhost` です。移行先の環境の運用管理サーバマシンのホスト名を指定した場合と、`@myhost` を指定した場合のホスト名の置換について、それぞれ説明します。

移行先の環境の運用管理サーバマシンのホスト名を指定する場合

移行先の環境の運用管理サーバマシンのホスト名を指定すると、出力する簡易構築定義ファイルで、次に示すタグの値が指定したホスト名に置換されます。

- ホストの定義の `<host-name>` タグ
- ユニットの定義の `<host-ref>` タグ
- 論理サーバのコンフィグレーションに指定されているパラメタのホスト名

@myhost を指定する場合

`@myhost` を指定すると、出力する簡易構築定義ファイルで、次に示すタグの値が `@myhost` に置換されます。

- ユニットの定義の `<host-ref>` タグ
- 論理サーバのコンフィグレーションに指定されているパラメタのホスト名

`@myhost` を指定した場合、ホストの定義は出力されません。ただし、`@myhost` を指定した場合に、ホストの定義の `<display-name>` タグ、`<description>` タグ、`<agent-host>` タグ、または `<agent-port>` タグに値を指定したときは、ホストの定義が出力されます。

-encoding <文字エンコーディング名>

出力する簡易構築定義ファイルの文字エンコーディングを指定します。指定を省略

8. Smart Composer 機能で使用するコマンド

した場合は UTF-8 が設定されます。

設定できる文字エンコーディングは、JavaVM がサポートしている文字エンコーディングとなります。JavaVM がサポートしている文字エンコーディングについては、JDK のドキュメントのサポートされているエンコーディングに関する説明を参照してください。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.encoding キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.encoding キーを省略している場合に、このオプションを省略してコマンドを実行したときは、UTF-8 が設定されます。

-o <出力ファイル>

出力する簡易構築定義ファイルの出力先を指定します。

入力例

```
<Cosminexusインストールディレクトリ>%manager%bin%cmx_export_model -o c:%temp%MyWebSystemexport.xml
```

出力形式

XML ファイル形式の簡易構築定義ファイルで出力します。簡易構築定義ファイルの出力例については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」を参照してください。

注意事項

Smart Composer 機能で構築したシステムに当てはまらない構成は、free-tier 構成で出力されます。出力した簡易構築定義ファイルに、free-tier の定義が含まれていた場合、出力した簡易構築定義ファイルに対して、Smart Composer 機能のコマンドでは、システム構築以外の操作は実行できません。

cmx_list_model (Web システムの情報モデルの取得)

形式

Web システムの情報モデルを取得する場合

```
cmx_list_model [共通引数] -modeltype  
{SERVER|APPLICATION|RESOURCE|RESADAPTER|RESJAVABEANS|SERVEREX}  
[{-s <Webシステム名>|-all|-r}]
```

Web システムに関連していない論理サーバの一覧を取得する場合

```
cmx_list_model [共通引数] -ns
```

機能

Web システムの情報モデルを取得する場合

登録されている Web システムの情報モデルを取得し、標準出力に取得結果を出力しま

す。

-s オプション, または .cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで Web システム名が指定されている場合は, その Web システムだけを対象とします。

Web システムに関連づいていない論理サーバの一覧を取得する場合

Web システムに関連づいていない論理サーバの一覧を取得し, 標準出力に取得結果を出力します。一覧には論理サーバの種別も表示します。

引数

-modeltype

表示する対象を指定します。

- SERVER

Web システムの情報モデル中の論理サーバの一覧を表示します。

- APPLICATION

Web システムの情報モデル中の J2EE アプリケーションの一覧を表示します。

- RESOURCE

Web システムの情報モデル中のリソースアダプタの一覧を表示します。

- RESADAPTER

Web システムの情報モデル中のリソースアダプタの一覧を表示します。

- RESJAVABEANS

Web システムの情報モデル中の JavaBeans リソースの一覧を表示します。

- SERVEREX

Web システムの情報モデル中の論理サーバの一覧を表示します。一覧には論理サーバの種別も表示します。

注 互換用の引数です。RESADAPTER を使用することをお勧めします。

-s <Web システム名>

Web システム名を指定します。

このオプションを省略した場合は, .cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。

-all

登録されているすべての Web システムを表示する対象とします。

-r

cmx_register_application コマンドで登録した J2EE アプリケーションの一覧, または cmx_register_resource コマンドで登録したリソースアダプタの一覧を表示する場合に指定します。

-ns

Web システムに関連づいていない論理サーバの一覧表示する場合に指定します。

出力形式

CSV 形式で出力します。

Web システムの情報モデルを取得する場合

- 論理サーバー一覧の場合 (SERVER 指定時)

```
<論理サーバ名>,<ホスト名>,<ティア種類> ,<サービスユニット名>,<Webシステム名>
```

注

論理ユーザサーバの場合,ティア種類には空文字が出力されます。

- 論理サーバー一覧の場合 (SERVEREX 指定時)

```
<論理サーバ名>,<論理サーバ種別>,<ホスト名>,<ティア種類> ,<サービスユニット名>,<Webシステム名>
```

注

論理ユーザサーバの場合,ティア種類には空文字が出力されます。

- Web システムに関連づけられたアプリケーション一覧の場合

```
<アプリケーション表示名>,<アプリケーションバージョン>,<Webシステム名>
```

- 登録されている J2EE アプリケーション一覧の場合

```
<アプリケーション表示名>,<アプリケーションバージョン>
```

- Web システムに関連づけられたリソースアダプター一覧の場合

```
<リソースアダプタ表示名>,<Webシステム名>
```

- 登録されているリソースアダプター一覧の場合

```
<リソースアダプタ表示名>
```

- Web システムに関連づけられた JavaBeans リソース一覧の場合

```
<JavaBeansリソース表示名>,<Webシステム名>
```

- 登録されている JavaBeans リソース一覧の場合

```
<JavaBeansリソース表示名>
```

- Web システムに関連づいていない論理サーバーの一覧

```
<論理サーバ名>,<論理サーバ種別>,<ホスト名>
```

入力例・出力例

- 論理サーバー一覧 (SERVER 指定時)

```
>cmx_list_model -modeltype SERVER
cmx_Web01_Unit01_PRF_01,Host01,combined-tier,Unit01,Web01
cmx_Web01_Unit01_J2EE_01,Host01,combined-tier,Unit01,Web01
cmx_Web01_Unit01_HTTP_01,Host01,combined-tier,Unit01,Web01
cmx_Web01_Unit02_PRF_01,Host01,combined-tier,Unit02,Web01
cmx_Web01_Unit02_J2EE_01,Host01,combined-tier,Unit02,Web01
cmx_Web01_Unit02_HTTP_01,Host01,combined-tier,Unit02,Web01
cmx_Web02_Unit01_PRF_01,Host01,combined-tier,Unit01,Web02
cmx_Web02_Unit01_USER_01,Host01,,Unit01,Web02
:
```

- 論理サーバー一覧 (SERVEREX 指定時)

```
>cmx_list_model -modeltype SERVER
cmx_Web01_Unit01_PRF_01,PRF,Host01,combined-tier,Unit01,Web01
cmx_Web01_Unit01_J2EE_01,J2EEServer,Host01,combined-tier,Unit01,
Web01
cmx_Web01_Unit01_HTTP_01,Webserver,Host01,combined-tier,Unit01,W
eb01
cmx_Web01_Unit02_PRF_01,PRF,Host01,combined-tier,Unit02,Web01
cmx_Web01_Unit02_J2EE_01,J2EEServer,Host01,combined-tier,Unit02,
Web01
cmx_Web01_Unit02_HTTP_01,Webserver,Host01,combined-tier,Unit02,W
eb01
cmx_Web02_Unit01_PRF_01,PRF,Host01,combined-tier,Unit01,Web02
cmx_Web02_Unit01_USER_01,UserServer,Host01,,Unit01,Web02
:
```

- Web システムに関連づけられたアプリケーション一覧

```
>cmx_list_model -modeltype APPLICATION
App01,Ver01,Web01
App02,Ver02,Web02
App03,Ver01,Web02
```

- 登録されているアプリケーション一覧

```
>cmx_list_model -modeltype APPLICATION -r
App01,Ver01
App02,Ver01
App02,Ver02
App03,Ver01
```

- Web システムに関連づけられたリソースアダプター一覧

```
>cmx_list_model -modeltype RESADAPTER
Rar01,Web01
Rar02,Web02
```

- 登録されているリソースアダプター一覧

```
>cmx_list_model -modeltype RESADAPTER -r
Rar01
Rar02
Rar03
Rar04
```

- Web システムに関連づけられた JavaBeans リソース一覧

8. Smart Composer 機能で使用するコマンド

```
>cmx_list_model -modeltype RESJAVABEANS
Jar01,Web01
Jar02,Web02
```

- 登録されている JavaBeans リソース一覧

```
>cmx_list_model -modeltype RESJAVABEANS -r
Jar01
Jar02
Jar03
Jar04
```

- Web システムに関連づいていない論理サーバー一覧

```
>cmx_list_model -ns
PRF_01,PRF,Host01
J2EE_01,J2EEServer,Host01
HTTP_01,Webserver,Host01
:
```

戻り値

0 :
正常終了しました。

1 :
警告終了しました。

64 :
異常終了しました。

注意事項

-s オプション, -all オプション, および .cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーの指定をすべて省略した場合は, 登録されているすべての Web システムを表示する対象とします。

.cmxrc ファイル, または cmxclient.properties ファイルの cmx.websystem.name キーに Web システム名を指定している場合に, すべての Web システムのモデルを取得するには, -all オプションを指定してください。

modeltype SERVER と -r を同時に指定した場合, エラーになります。

統合ネーミングスケジューラサーバ用の Web システムの場合, -modeltype オプションに APPLICATION, RESOURCE, RESADAPTER, または RESJAVABEANS を指定すると, 何も表示されません。

cmx_list_status (サービスユニット状況の表示)

形式

Web システム中のサービスユニットのステータスを取得する場合

```
cmx_list_status [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]
```

Web システムに関連づいていない論理サーバのステータスを取得する場合

```
cmx_list_status [共通引数] -ns
```

機能

Web システム中のサービスユニットのステータスを取得する場合

サービスユニットの稼働状況のステータス、およびサービスユニットに対する cmx_build_system コマンドの実行状態を取得し、標準出力に取得結果を出力します。

サービスユニットの稼働状況のステータス

サービスユニットの状態を取得し、標準出力に取得結果を出力します。出力されるサービスユニット稼働状況のステータスは次の 5 種類です。

- working

稼働状態を表します。サービスユニット中の全論理サーバが稼働中で、負荷分散機からリクエストが振り分けられる状態です。

- ready

準備状態を表します。サービスユニット中の全論理サーバが稼働中で、負荷分散機からのリクエスト振り分けが抑止されている状態です。

通常、アプリケーションのテストなどは、停止状態から準備状態にして、アプリケーションのデプロイや外部との接続をしないで実施します。

- hold

閉塞状態を表します。サービスユニット中の Web サーバが計画停止済み（ほかの論理サーバは稼働中）で、負荷分散機からのリクエスト振り分けが抑止されている状態です。通常、J2EE アプリケーションの入れ替えなどは、稼働状態から閉塞状態にして、実施します。

なお、閉塞状態と準備状態との違いは、Web サーバが起動しているかどうかだけです。

- stopped

停止状態を表します。サービスユニット中の全論理サーバが停止済みで、負荷分散機からのリクエスト振り分けが抑止されている状態です。

- unknown

障害発生や未セットアップのために状態の判別がつかない状態を表します。

cmx_build_system の実行状態

サービスユニットに対する cmx_build_system の実行状態を取得し、標準出力に done（実行済み）と not done（未実行）で取得結果を出力します。

8. Smart Composer 機能で使用するコマンド

コマンドのオプションで、サービスユニットが指定されていれば特定のサービスユニットが、省略されていれば Web システム中の全ユニットが表示の対象となります。

なお、それぞれの J2EE アプリケーションやそれぞれの論理サーバの稼働状態を知りたい場合は、mngsvrutil コマンドを使用してください。mngsvrutil コマンドについては、「7.2 Management Server で使用するコマンドの詳細」の「mngsvrutil (Management Server の運用管理コマンド)」を参照してください。

Web システムに関連づいていない論理サーバのステータスを取得する場合

論理サーバの稼働状況のステータス

論理サーバの状態を取得し、標準出力に取得結果を出力します。出力されるサービスユニット稼働状況のステータスについては、「7.3 mngsvrutil コマンドのサブコマンドの詳細」を参照してください。

論理サーバの設定情報の配布の実行状態

論理サーバに対する設定情報の配布の実行状態を取得し、標準出力に done (実行済み) と not done (未実行) で取得結果を出力します。設定情報の配布が実行済みの状態とは、次の 2 種類の状態を指します。

- cmx_build_system コマンドを引数に -ns を指定して実行し、正常にシステムの構築が完了した状態
- 運用管理ポータルで正常に論理サーバの配布が完了した状態

引数

-s <Web システム名>

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-unit <サービスユニット名>

サービスユニット名を指定します。

-ns

Web システムに関連づいていない論理サーバのステータスを取得する場合に指定します。

出力形式

CSV 形式で出力されます。

Web システム中のサービスユニットのステータスを表示する場合

```
<サービスユニット名>,<Web システム名>,<cmx_build_system 実行状態>,<サービスユニットの稼働状態>
```

Web システムに関連づいていない論理サーバのステータスを表示する場合

```
<論理サーバ名>,<論理サーバの設定情報の配布の実行状態> ,<論理サーバの稼働状態>
```

注 論理サーバの種類が J2EE サーバ、SFO サーバ、Web サーバ以外の場合は論理サーバの設定情報の配布の実行状態は空文字で表示されます。

入力例・出力例

Web システム中のサービスユニットのステータスを表示する場合

```
>cmx_list_status -s sample-system
unit01,sample-system,done,working
unit02,sample-system,done,ready
unit03,sample-system,done,hold
unit04,sample-system,done,stopped
unit05,sample-system,done,unknown
unit06,sample-system,not done,unknown
```

Web システムに関連づいていない論理サーバのステータスを表示する場合

```
>cmx_list_status -ns
PRF01,,working
J2EE01,done,working
HTTP01,done,working
```

戻り値

- 0 :
正常終了しました。
- 1 :
警告終了しました。
- 64 :
異常終了しました。

注意事項

メモリセッションフェイルオーバー用のサービスユニットの場合、サービスユニットの稼働状況のステータスには、「working」、「stopped」および「unknown」の3種類が表示されます。

CTM 用のサービスユニットおよび統合ネーミングスケジューラサーバ用のサービスユニットの場合、サービスユニットの稼働状況のステータスには、「working」、「stopped」および「unknown」の3種類が表示されます。

CTM 用のサービスユニットに複数の J2EE サーバを配置している場合に、サービスユニットが正常に起動したあと、一部の J2EE サーバが異常停止したときは、サービ

スユニットの稼働状況のステータスには「unknown」が表示されます。

cmx_register_application (J2EE アプリケーションの登録) (互換用)

形式

- アーカイブ形式の J2EE アプリケーションを登録する場合

```
cmx_register_application [共通引数] -f <EARファイル> -appver <アプリケーションバージョンID> [-replace]
```

- 展開ディレクトリ形式の J2EE アプリケーションを登録する場合

```
cmx_register_application [共通引数] -appname <アプリケーション表示名> -d <アプリケーションディレクトリ> [-replace]
```

機能

Management Server が管理するドメインに J2EE アプリケーションを登録します。

登録できる J2EE アプリケーションの形式には、アーカイブ形式および展開ディレクトリ形式があります。

- アーカイブ形式の J2EE アプリケーションを登録する場合
アーカイブ形式の J2EE アプリケーションは、アプリケーション表示名とアプリケーションバージョン ID で一意に識別されます。同じアプリケーション表示名、同じアプリケーションバージョン ID のアプリケーションがすでに登録されている場合は、`-replace` オプションを指定することによって再登録できます。
- 展開ディレクトリ形式の J2EE アプリケーションを登録する場合
展開ディレクトリ形式の J2EE アプリケーションは、アプリケーション表示名で一意に識別されます。同じアプリケーション表示名のアプリケーションがすでに登録されている場合は、`-replace` オプションを指定することによって再登録できます。

引数

`-f <EAR ファイル>`

J2EE アプリケーションの EAR ファイルのパスを指定します。

`-appver <アプリケーションバージョン ID>`

J2EE アプリケーションのバージョン ID を指定します。

`-appname <アプリケーション表示名>`

アプリケーションディレクトリ下の DD (application.xml) の display-name タグに記述したアプリケーション表示名を指定します。

`-d <アプリケーションディレクトリパス>`

アプリケーションサーバマシン (J2EE サーバ) から参照できるアプリケーション

ディレクトリのパスを指定します。

-replace

J2EE アプリケーションを再登録する場合に指定します。

入力例

```
cmx_register_application -f App1.ear -appver 1.1 -replace
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

J2EE アプリケーションを登録する形式が異なっても、同じアプリケーション表示名のアプリケーションは登録できません。

また、同じアプリケーション表示名のアプリケーションがすでに登録されている場合に、-replace オプションを指定しても、登録するアプリケーションの形式が異なるときは再登録できません。異なる形式のアプリケーションを再登録する場合は、登録済みのアプリケーションを削除してから登録してください。

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_register_resource (リソースの登録)(互換用)

形式

- リソースアダプタを登録する場合

```
cmx_register_resource [共通引数] -f <RARファイル> [-c <Connector属性ファイル>] [-replace]
```

- JavaBeans リソースを登録する場合

```
cmx_register_resource [共通引数] -f <JARファイル> -c <JavaBeansリソース属性ファイル> [-replace]
```

機能

Management Server が管理するドメインにリソースアダプタまたは JavaBeans リソースを登録します。

8. Smart Composer 機能で使用するコマンド

-f オプションの引数で指定したリソースアダプタまたは JavaBeans リソースの表示名と同じ表示名のリソースがすでに登録されている場合は、-replace オプションを指定することによって、再登録できます。

引数

-f <RAR ファイルまたは JAR ファイル>

登録するリソースアダプタの RAR ファイルまたは JavaBeans リソースの JAR ファイルのパスを指定します。

-c <Connector 属性ファイルまたは JavaBeans リソース属性ファイル>

登録するリソースアダプタの Connector 属性ファイルまたは JavaBeans リソース属性ファイルのパスを指定します。JavaBeans リソースを登録する場合は、-c オプションで JavaBeans リソース属性ファイルを必ず指定してください。

-replace

リソースを再登録する場合に指定します。

入力例

```
cmx_register_resource -f App1.rar -replace
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

RAR ファイルと Connector 属性ファイルで異なる表示名が指定された場合、Connector 属性ファイルの表示名で登録します。リソースアダプタと JavaBeans リソースで同じ表示名が指定された場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_resume_lb (負荷分散機の設定再開始)

形式

```
cmx_resume_lb [共通引数] [-s <Webシステム名>]
```

機能

負荷分散機を冗長化したシステムの場合に、障害の発生した負荷分散機の休止状態を解除して、設定処理を再開します。

負荷分散機を冗長化したシステムの場合、`cmx_build_system` コマンドなどを入力すると、2 台の負荷分散機に対して同じ設定が同時に実行されます。一方の負荷分散機で障害が発生すると、障害が発生した負荷分散機に対する設定は抑止され、もう一方の負荷分散機だけに設定を実行する運用になります。この場合、障害の発生した負荷分散機を正常な負荷分散機と同じ設定に復旧してから、`cmx_resume_lb` コマンドを実行します。`cmx_resume_lb` コマンドが正常に受け付けられると、両方の負荷分散機に対して同じ設定を同時に実行する運用に回復されます。

引数

`-s <Web システム名 >`

Web システム名を指定します。

このオプションを省略した場合は、`.cmxrc` ファイルまたは `cmxclient.properties` ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

入力例

```
cmx_resume_lb -s sample-system
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

負荷分散機を冗長化したシステムでない場合、または負荷分散機が休止状態でない場合はエラーになります。

cmx_scaleout_host (Web システムの情報モデルの変更)

形式

```
cmx_scaleout_host [共通引数] -f <スケールアウト用ホスト定義ファイル>
```

機能

スケールアウト用ホスト定義ファイルの内容に従って、ホスト単位管理モデルの Web システムの情報モデルを変更します。JP1/SC/DPM を利用して、ホスト単位管理モデルの Web システムをスケールアウトする場合に使用します。このコマンドは複製先のホストで実行します。

引数

-f <スケールアウト用ホスト定義ファイル>

スケールアウト用ホスト定義ファイルのパスを指定します。

入力例

```
cmx_scaleout_host -f MyWebSystemHostScaleout.xml
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

運用管理サーバモデルの Web システムのホストに対して、このコマンドを実行した場合はエラーになります。

負荷分散機および CTM を使用したシステムで、このコマンドを実行する場合の注意事項を次に示します。

- 負荷分散機を使用したシステムの場合

複製元のホストと複製先のホストは、負荷分散機の仮想サーバを共有します。負荷分散機で使用するサーバ ID の割り振りルールの範囲が、ホスト間で重複しないように、スケールアウト用ホスト定義ファイルを設定してください。

- CTM を使用したシステムの場合

複製元のホストと複製先のホストは、同じ CTM ドメインに属します。複製元のホストに CTM 識別子が定義されている場合、複製先のホストにも同じ CTM 識別子が定義されるため、CTM ドメイン内で CTM 識別子が重複することになります。このため、複製元のホストでは、CTM 識別子の設定を省略することをお勧めします。CTM 識別子の設定を省略した場合、CTM 識別子は、ホスト間で一意になるように、Management Server で自動的に設定されます。

なお、複製元のホストで CTM 識別子を定義している場合は、このコマンドを実行したあとに、複製先のホストの CTM 識別子を CTM ドメイン内でユニークになる

ように設定し直してください。

cmx_start_application (J2EE アプリケーションの開始)(互換用)

形式

- J2EE アプリケーションを指定して開始する場合

```
cmx_start_application [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -appname <アプリケーション表示名>
```

- J2EE アプリケーションを指定しないで開始する場合

```
cmx_start_application [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]
```

機能

J2EE アプリケーションを開始します。コマンドに J2EE アプリケーションを指定している場合と、指定していない場合とで開始対象となる J2EE アプリケーションが異なります。

- J2EE アプリケーションを指定する場合

Web システムに含まれるすべてのサービスユニット上で、指定された J2EE アプリケーションを開始します。

サービスユニットが指定されている場合は、指定されたサービスユニット上で、指定された J2EE アプリケーションを開始します。

- J2EE アプリケーションを指定していない場合

Web システムに含まれるすべてのサービスユニット上で、その Web システムに関連づけられているすべての J2EE アプリケーション (アーカイブ形式および展開ディレクトリ形式) を開始します。

サービスユニットが指定されている場合は、指定されたサービスユニット上で、その Web システムに関連づけられているすべての J2EE アプリケーションを開始します。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-unit < サービスユニット名 >

サービスユニット名を指定します。

8. Smart Composer 機能で使用するコマンド

`-appname <アプリケーション表示名>`

J2EE アプリケーションの表示名を指定します。 `cmx_register_application` コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_start_application -s sample-system -unit unit1 -appname Appl
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_start_resource (リソースの開始)(互換用)

形式

- リソースアダプタを指定して開始する場合

```
cmx_start_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -resname <リソースアダプタ表示名>
```

- JavaBeans リソースを指定して開始する場合

```
cmx_start_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -resname <JavaBeansリソース表示名>
```

- リソースアダプタまたは JavaBeans リソースを指定しないで開始する場合

```
cmx_start_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]
```

機能

リソースアダプタまたは JavaBeans リソースを開始します。コマンドにリソースアダプタまたは JavaBeans リソースを指定している場合と、指定していない場合とで開始対象となるリソースアダプタまたは JavaBeans リソースが異なります。

- リソースアダプタまたは JavaBeans リソースを指定している場合
Web システムに含まれるすべてのサービスユニット上で、指定されたリソースアダプタまたは JavaBeans リソースを開始します。
サービスユニットが指定されている場合は、指定されたサービスユニット上で、指定されたリソースアダプタまたは JavaBeans リソースを開始します。
- リソースアダプタまたは JavaBeans リソースを指定していない場合
Web システムに含まれるすべてのサービスユニット上で、その Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースを開始します。
サービスユニットが指定されている場合は、指定されたサービスユニット上で、その Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースを開始します。

引数

`-s <Web システム名 >`

Web システム名を指定します。

このオプションを省略した場合は、`.cmxrc` ファイルまたは `cmxclient.properties` ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

`-unit <サービスユニット名 >`

サービスユニット名を指定します。

`-resname <リソースアダプタ表示名または JavaBeans リソース表示名 >`

リソースアダプタまたは JavaBeans リソースの表示名を指定します。

`cmx_register_resource` コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_start_resource -s sample-system -unit unit1 -resname Appl
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

コネクシオンプールのクラスタ化機能を利用しているリソースアダプタが開始対象に含まれている場合は、必ず `-resname` オプションでリソースアダプタを指定して、一つずつ開始してください。この場合、メンバリソースアダプタを開始したあとに、ルートリソースアダプタを開始してください。

また、コネクシオンプールのクラスタ化機能を利用しているリソースアダプタと、利用していないリソースアダプタとが開始対象に含まれている場合は、コネクシオンプールのクラスタ化機能を利用しているリソースアダプタをすべて開始したあと、`-resname` オプションを指定しない `cmx_start_resource` コマンドを入力して、ほかのリソースアダプタを開始してください。

データベース上のキューに接続するためのリソースアダプタが開始対象に含まれている場合は、必ず `-resname` オプションでリソースアダプタを指定して、次の順序で開始してください。

1. DB Connector for Cosminexus RM
2. Cosminexus RM

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_start_target (Web システムまたはサービスユニットの開始)

形式

Web システムを開始する場合

```
cmx_start_target [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -mode { ALL|READY } [-strict]
```

Web システムに関連づいていない論理サーバを含むシステムを開始する場合

```
cmx_start_target [共通引数] -ns
```

機能

システムの開始をします。Web システムを開始する場合と、Web システムに関連づいていない論理サーバを含むシステムを開始する場合で動作が異なります。それぞれの場合の動作について説明します。

Web システムを開始する場合

指定されたモードに応じて、開始対象のサービスユニットの状態を稼働状態または準備状態にします。稼働状態とは、サービスユニット中の全論理サーバが稼働中で負荷分散機からリクエストが振り分けられる状態です。準備状態とは、サービスユニット

中の全論理サーバが稼働中で、負荷分散機からのリクエスト振り分けが抑止されている状態です。

コマンドのオプションで、サービスユニットが指定されていれば特定のサービスユニットが、省略されていれば Web システム中の全サービスユニットが、開始の対象となります。

サービスユニットの状態は、`cmx_list_status` コマンドで参照できます。サービスユニットの状態の種類、およびサービスユニットの状態の参照方法については、「`cmx_list_status`」を参照してください。

Web システムに関連していない論理サーバを含むシステムを開始する場合、開始対象のシステムに含まれるすべての論理サーバを起動して、稼働状態にします。すでに稼働中の論理サーバは、引き続き稼働状態が保たれ、この操作によって再起動されることはありません。

運用管理ポータルで論理サーバの起動順序を設定している場合は、その順序に従って起動されます。

Web システムに関連していない論理サーバの状態は、`cmx_list_status` コマンドに `-ns` オプションを指定することで参照できます。論理サーバの状態の種類、および論理サーバの状態の参照方法については、「`cmx_list_status`」を参照してください。

引数

`-s <Web システム名 >`

Web システム名を指定します。

このオプションを省略した場合は、`.cmxrc` ファイルまたは `cmxclient.properties` ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

`-unit <サービスユニット名 >`

サービスユニット名を指定します。

`-mode { ALL|READY }`

開始モードを指定します。

- ALL

停止状態、閉塞状態、準備状態を稼働状態に引き上げます。

- READY

停止状態、閉塞状態を準備状態に引き上げます。

稼働状態のサービスユニットを準備状態に引き下げることはできません。

`-strict`

論理サーバの起動に失敗した場合に、以降の論理サーバの起動を中断するときに指定します。

このオプションを省略してコマンドを実行した場合、論理サーバの起動に失敗した

8. Smart Composer 機能で使用するコマンド

とき、以降の論理サーバの起動を継続します。

論理サーバの起動に失敗した場合に、以降の論理サーバの起動を継続するか、中断するかを選択については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」のサービスユニットの起動の説明を参照してください。

-ns

Web システムに関連づいていない論理サーバを含むシステムを起動する場合に指定します。

入力例

```
cmx_start_target -s sample-system -unit unit1 -mode ALL
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

戻り値の詳細を次の表に示します。

戻り値	サービスユニットが一つの場合	サービスユニットが複数の場合
0	すべての論理サーバの起動に成功しました。	すべてのサービスユニットの起動に成功 ¹ しました。
1	-	一部のサービスユニットの起動に失敗 ² しました。
64	一部の論理サーバの起動に失敗しました。	すべてのサービスユニットの起動に失敗 ² しました。

(凡例) - : なし。

注 1 サービスユニットの起動成功とは、サービスユニット中のすべての論理サーバの起動に成功したことです。

注 2 サービスユニットの起動失敗とは、サービスユニット中の一部の論理サーバの起動に失敗したことです。

注意事項

-unit オプションでメモリセッションフェイルオーバー用、CTM 用または統合ネーミングスケジューラサーバ用のサービスユニットを指定する場合、-mode オプションに READY を指定したときはエラーになります。-unit オプションを省略している場合に、-mode オプションに READY を指定した場合、Web システムに属するメモリセッションフェイルオーバー用、CTM 用および統合ネーミングスケジューラサーバ用のサー

ビスユニットの状態は、稼働状態に引き上げられます。

開始対象のサービスユニットに含まれている論理ユーザサーバは、`-mode` オプションの指定に関係なく、起動されます。開始対象のサービスユニットに論理ユーザサーバが含まれている場合に、論理ユーザサーバの起動に失敗したときは、サービスユニットの開始も失敗したとみなされます。この場合、`-unit` オプションを指定したコマンドは異常終了し、`-unit` オプションを省略したコマンドは警告終了します。

複数の J2EE サーバを配置した CTM 用のサービスユニットを起動する場合、一部の J2EE サーバで起動処理を失敗しても、すべての J2EE サーバに対して起動処理を行います。この場合、コマンドは異常終了します。

cmx_stop_application (J2EE アプリケーションの停止)(互換用)

形式

- J2EE アプリケーションを指定して停止する場合

```
cmx_stop_application [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -appname <アプリケーション表示名>
```

- J2EE アプリケーションを指定しないで停止する場合

```
cmx_stop_application [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]
```

機能

J2EE アプリケーションを停止します。コマンドに J2EE アプリケーションを指定している場合と、指定していない場合とで停止対象となる J2EE アプリケーションが異なります。

- J2EE アプリケーションを指定する場合

Web システムに含まれるすべてのサービスユニット上で、指定された J2EE アプリケーションを停止します。

サービスユニットが指定されている場合は、指定されたサービスユニット上で、指定された J2EE アプリケーションを停止します。

- J2EE アプリケーションを指定していない場合

Web システムに含まれるすべてのサービスユニット上で、その Web システムに関連づけられているすべての J2EE アプリケーション (アーカイブ形式および展開ディレクトリ形式) を停止します。

サービスユニットが指定されている場合は、指定されたサービスユニット上で、その Web システムに関連づけられているすべての J2EE アプリケーションを停止します。

引数

`-s <Web システム名 >`

Web システム名を指定します。

このオプションを省略した場合は、`.cmxrc` ファイルまたは `cmxclient.properties` ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

`-unit < サービスユニット名 >`

サービスユニット名を指定します。

`-appname < アプリケーション表示名 >`

J2EE アプリケーションの表示名を指定します。`cmx_register_application` コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_stop_application -s sample-system -unit unit1 -appname App1
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_stop_resource (リソースの停止)(互換用)

形式

- リソースアダプタを指定して停止する場合

```
cmx_stop_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -resname <リソースアダプタ表示名>
```

- JavaBeans リソースを指定して停止する場合
`cmx_stop_resource` [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -resname <JavaBeansリソース表示名>
- リソースアダプタまたは JavaBeans リソースを指定しないで停止する場合
`cmx_stop_resource` [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]

機能

リソースアダプタまたは JavaBeans リソースを停止します。コマンドにリソースアダプタまたは JavaBeans リソースを指定している場合と、指定していない場合とで停止対象となるリソースアダプタまたは JavaBeans リソースが異なります。

- リソースアダプタまたは JavaBeans リソースを指定している場合
 Web システムに含まれるすべてのサービスユニット上で、指定されたリソースアダプタまたは JavaBeans リソースを停止します。
 サービスユニットが指定されている場合は、指定されたサービスユニット上で、指定されたリソースアダプタまたは JavaBeans リソースを停止します。
- リソースアダプタまたは JavaBeans リソースを指定していない場合
 Web システムに含まれるすべてのサービスユニット上で、その Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースを停止します。
 サービスユニットが指定されている場合は、指定されたサービスユニット上で、その Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースを停止します。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、`.cmxrc` ファイルまたは `cmxclient.properties` ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-unit <サービスユニット名 >

サービスユニット名を指定します。

-resname <リソースアダプタ表示名または JavaBeans リソース表示名 >

リソースアダプタまたは JavaBeans リソースの表示名を指定します。

`cmx_register_resource` コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_stop_resource -s sample-system -unit unit1 -resname App1
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

コネクションプールのクラスタ化機能を利用しているリソースアダプタが停止対象に含まれている場合は、必ず `-resname` オプションでリソースアダプタを指定して、一つずつ停止してください。この場合、ルートリソースアダプタを停止したあとに、メンバリソースアダプタを停止してください。

また、コネクションプールのクラスタ化機能を利用しているリソースアダプタと、利用していないリソースアダプタとが停止対象に含まれている場合は、コネクションプールのクラスタ化機能を利用しているリソースアダプタをすべて停止したあと、`-resname` オプションを指定しない `cmx_stop_resource` コマンドを入力して、ほかのリソースアダプタを停止してください。

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_stop_target (Web システムまたはサービスユニットの停止)

形式

Web システムを停止する場合

```
cmx_stop_target [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -mode { ALL|HOLD }
```

Web システムに関連づいていない論理サーバを含むシステムを停止する場合

```
cmx_stop_target [共通引数] -ns [-normal]
```

機能

Web システムを停止する場合

指定されたモードに応じて、停止対象のサービスユニットの状態を停止状態または閉

塞状態にします。停止状態とは、サービスユニット中の全論理サーバが停止済みで、負荷分散機からのリクエスト振り分けが抑止されている状態です。また、閉塞状態とは、サービスユニット中の Web サーバが計画停止済み（ほかの論理サーバは稼働中）で、負荷分散機からのリクエスト振り分けが抑止されている状態です。コマンドのオプションで、サービスユニットが指定されていれば特定のサービスユニットが、省略されていれば Web システム中の全サービスユニットが停止の対象となります。

なお、このコマンドでは、実行中のクライアントからのリクエストを中断させないために、Web サーバは計画停止させます。

サービスユニットの状態は、`cmx_list_status` コマンドで参照できます。サービスユニットの状態の種類、およびサービスユニットの状態の参照方法については、「`cmx_list_status`」を参照してください。

サービスユニット停止時の動作については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の Web システムに含まれているサービスユニットを停止する場合の説明を参照してください。

Web システムに関連づいていない論理サーバを含むシステムを停止する場合開始対象のシステムに含まれるすべての論理サーバを停止します。すでに停止中の論理サーバは、引き続き停止状態が保たれます。

運用管理ポータルで論理サーバの起動順序を設定している場合は、その順序の逆順に従って停止されます。また、Web サーバは計画停止します。計画停止とは、現在受け付けているすべてのリクエストの処理終了を待って論理サーバを停止させることです。Web サーバの計画停止をしたくない場合は `-normal` オプションを指定します。

Web システムに関連づいていない論理サーバの状態は、`cmx_list_status` コマンドに `-ns` オプションを指定することで参照できます。論理サーバの状態の種類、および論理サーバの状態の参照方法については、「`cmx_list_status`」を参照してください。

サービスユニット停止時の動作については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の Web システムに関連づいていない論理サーバを含むシステムを停止する場合の説明を参照してください。

引数

`-s <Web システム名>`

Web システム名を指定します。

このオプションを省略した場合は、`.cmxrc` ファイルまたは `cmxclient.properties` ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

`-unit <サービスユニット名>`

サービスユニット名を指定します。

8. Smart Composer 機能で使用するコマンド

`-mode { ALL | HOLD }`

停止モードを指定します。

- ALL

稼働状態、準備状態、閉塞状態を停止状態に引き下げます。

- HOLD

稼働状態、準備状態を閉塞状態に引き下げます。

停止状態のサービスユニットを閉塞状態に引き上げることはできません。

`-ns`

Web システムに関連していない論理サーバを含むシステムを停止する場合に指定します。

`-normal`

Web サーバの計画停止をしたくない場合に指定します。このオプションを指定しないでコマンドを実行すると、Web サーバは計画停止します。

入力例

```
cmx_stop_target -s sample-system -unit unit1 -mode ALL
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

`-unit` オプションでメモリセッションフェイルオーバー用、CTM 用または統合ネーミングスケジューラサーバ用のサービスユニットを指定する場合に、`-mode` オプションに `HOLD` を指定したときはエラーになります。`-unit` オプションを省略している場合に、`-mode` オプションに `HOLD` を指定した場合、Web システム中のメモリセッションフェイルオーバー用、CTM 用および統合ネーミングスケジューラサーバ用のサービスユニットは、稼働状態のままで変更されません。

停止対象のサービスユニットに論理ユーザサーバが含まれている場合に、`-mode` オプションに `HOLD` を指定したときは、サービスユニットは閉塞状態となりますが、論理ユーザサーバが停止されません。また、停止対象のサービスユニットに論理ユーザサーバが含まれている場合に、`-mode` オプションに `ALL` を指定したときは、論理ユーザサーバが停止されます。論理ユーザサーバの停止に失敗した場合は、サービスユニットの停止も失敗したとみなされます。この場合、`-unit` オプションを指定したコマンドは異常終了し、`-unit` オプションを省略したコマンドは警告終了します。

複数の J2EE サーバを配置した CTM 用のサービスユニットを停止する場合、一部の J2EE サーバで停止処理を失敗しても、すべての J2EE サーバに対して停止処理を行います。この場合、コマンドは異常終了します。

cmx_test_lb (負荷分散機との接続テスト)

形式

```
cmx_test_lb [共通引数] [-s <Webシステム名>]
```

機能

引数で指定された Web システムに定義している負荷分散機を操作できるか、テストします。

このコマンドを利用することで、簡易構築定義ファイルの負荷分散機の定義内容や Smart Composer 機能の環境設定、および JP1/Cm2/NC の環境設定が正しく設定されているかどうか確認できます。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

入力例

```
cmx_test_lb -s sample-system
```

戻り値

0 :

負荷分散機の接続に成功しました。

1 :

指定された Web システムには負荷分散機が定義されていません。

64 :

負荷分散機との接続に失敗しました。

cmx_trans_param (抽象パラメタの展開)

形式

```
cmx_trans_param -i <入力ファイル> -p <抽象パラメタ指定> -o <出力ファイル>
```

機能

入力した簡易構築定義ファイル，または構成変更定義ファイル（論理サーバのパラメタ変更用）に対して，指定された抽象パラメタを展開し，Web システムに適した物理ティアのコンフィグレーションを生成します。

参考

Web システムを初期構築する場合は簡易構築定義ファイルを，構築した Web システムの構成を変更する（論理サーバのパラメタを変更する）場合は構成変更定義ファイルを使用します。

なお，構成変更定義ファイルには，論理サーバのパラメタ変更用と，サービスユニット・ホスト追加用の 2 種類があります。このコマンドは Web システムの初期構築をする場合，および Web システム全体の設定を変更する場合に使用するため，サービスユニット・ホスト追加用の構成変更定義ファイルは指定できません。

引数

-i <入力ファイル>

簡易構築定義ファイル，または構成変更定義ファイル（論理サーバのパラメタ変更用）を指定します。

-p <抽象パラメタ指定>

<抽象パラメタ指定> には，<抽象パラメタ>=<値> の形式で，抽象パラメタとその値を指定します。

抽象パラメタにはリクエスト最大同時処理数を指定します。リクエスト最大同時処理数のパラメタ名は「UnitMaxParallelCount」であるため，<抽象パラメタ指定> には，「UnitMaxParallelCount=<値>」を指定します。

なお，抽象パラメタの詳細については，「付録 I Smart Composer 機能で使用できる抽象パラメタ」を参照してください。

-o <出力ファイル>

抽象パラメタ展開後の出力ファイルを指定します。

入力例

```
cmx_trans_param -i MyWebSystemDef.xml -p UnitMaxParallelCount=10 -o outMyWebSystemDef.xml
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

<入力ファイル> に指定した物理ティアが combined-tier 以外の場合はエラーになります。

cmx_undefine_application (J2EE アプリケーションと Web システムの関連づけ解除)(互換用)

形式

- アーカイブ形式の J2EE アプリケーションの関連づけを削除する場合

```
cmx_undefine_application [共通引数] [-s <Webシステム名>] -appname <アプリケーション表示名> -appver <アプリケーションバージョンID>
```

- 展開ディレクトリ形式の J2EE アプリケーションの関連づけを削除する場合

```
cmx_undefine_application [共通引数] [-s <Webシステム名>] -appname <アプリケーション表示名>
```

機能

J2EE アプリケーションと Web システムの関連づけを削除します。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-appname < アプリケーション表示名 >

J2EE アプリケーションの表示名を指定します。cmx_register_application コマンド実行後にメッセージに出力される表示名を使用します。

-appver < アプリケーションバージョン ID >

J2EE アプリケーションのバージョン ID を指定します。

入力例

```
cmx_undefine_application -s sample-system -appname App1 -appver 1.0
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

指定した J2EE アプリケーションがデプロイされている場合はエラーとなります。

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_undefine_resource (リソースと Web システムの関連づけ解除)(互換用)

形式

- リソースアダプタの関連づけを削除する場合

```
cmx_undefine_resource [共通引数] [-s <Webシステム名>] -resname <リソースアダプタ表示名>
```

- JavaBeans リソースの関連づけを削除する場合

```
cmx_undefine_resource [共通引数] [-s <Webシステム名>] -resname <JavaBeans リソース表示名>
```

機能

リソースアダプタまたは JavaBeans リソースと Web システムの関連づけを削除します。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

`-resname` <リソースアダプタ表示名または JavaBeans リソース表示名>
 リソースアダプタまたは JavaBeans リソースの表示名を指定します。
`cmx_register_resource` コマンド実行後にメッセージに出力される表示名を使用し
 ます。

入力例

```
cmx_undefine_resource -s sample-system -resname App1
```

戻り値

0 :
 正常終了しました。

1 :
 警告終了しました。

64 :
 異常終了しました。

注意事項

指定したリソースアダプタまたは JavaBeans リソースがデプロイされている場合はエ
 ラーとなります。

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコ
 マンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_undeploy_application (J2EE アプリケーションのアンデ プロイ)(互換用)

形式

- J2EE アプリケーションを指定してアンデプロイする場合
`cmx_undeploy_application` [共通引数] [-s <Webシステム名>] [-unit <サー
 ビスユニット名>] -appname <アプリケーション表示名>
- J2EE アプリケーションを指定しないでアンデプロイする場合
`cmx_undeploy_application` [共通引数] [-s <Webシステム名>] [-unit <サー
 ビスユニット名>]

機能

J2EE アプリケーションをアンデプロイします。コマンドに J2EE アプリケーションを
 指定している場合と、指定していない場合とでアンデプロイのしかたが異なります。

- J2EE アプリケーションを指定する場合
 Web システムに含まれるすべてのサービスユニットから、指定された J2EE アプリ

8. Smart Composer 機能で使用するコマンド

ケーションをアンデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットから、指定された J2EE アプリケーションをアンデプロイします。

- J2EE アプリケーションを指定しない場合

Web システムに含まれるすべてのサービスユニットから、その Web システムに関連づけられているすべての J2EE アプリケーション（アーカイブ形式および展開ディレクトリ形式）をアンデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットから、その Web システムに関連づけられているすべての J2EE アプリケーションをアンデプロイします。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties ファイルの cmx.websystem.name キーで設定した値が仮定されます。なお、.cmxrc ファイルおよび cmxclient.properties ファイルの cmx.websystem.name キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

-unit < サービスユニット名 >

サービスユニット名を指定します。

-appname < アプリケーション表示名 >

J2EE アプリケーションの表示名を指定します。cmx_register_application コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_undeploy_application -s sample-system -unit unit1 -appname App1
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

注意事項

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_undeploy_resource (リソースのアンデプロイ)(互換用)

形式

- リソースアダプタを指定してアンデプロイする場合

```
cmx_undeploy_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -resname <リソースアダプタ表示名>
```

- JavaBeans リソースを指定してアンデプロイする場合

```
cmx_undeploy_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>] -resname <JavaBeansリソース表示名>
```

- リソースアダプタまたは JavaBeans リソースを指定しないでアンデプロイする場合

```
cmx_undeploy_resource [共通引数] [-s <Webシステム名>] [-unit <サービスユニット名>]
```

機能

リソースアダプタまたは JavaBeans リソースをサービスユニットからアンデプロイします。コマンドにリソースアダプタまたは JavaBeans リソースを指定している場合と、指定していない場合とでアンデプロイのしかたが異なります。

- リソースアダプタまたは JavaBeans リソースを指定している場合

Web システムに含まれるすべてのサービスユニットから、指定されたリソースアダプタまたは JavaBeans リソースをアンデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットから、指定されたリソースアダプタまたは JavaBeans リソースをアンデプロイします。

- リソースアダプタまたは JavaBeans リソースを指定していない場合

Web システムに含まれるすべてのサービスユニットから、その Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースをアンデプロイします。

サービスユニットが指定されている場合は、指定したサービスユニットから、その Web システムに関連づけられているすべてのリソースアダプタまたは JavaBeans リソースをアンデプロイします。

引数

-s <Web システム名 >

Web システム名を指定します。

このオプションを省略した場合は、.cmxrc ファイルまたは cmxclient.properties

8. Smart Composer 機能で使用するコマンド

ファイルの `cmx.websystem.name` キーで設定した値が仮定されます。なお、`.cmxrc` ファイルおよび `cmxclient.properties` ファイルの `cmx.websystem.name` キーを省略している場合に、このオプションを省略してコマンドを実行したときは、エラーになります。

`-unit <サービスユニット名>`

サービスユニット名を指定します。

`-resname <リソースアダプタ表示名または JavaBeans リソース表示名>`

リソースアダプタまたは JavaBeans リソースの表示名を指定します。

`cmx_register_resource` コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_undeploy_resource -s sample-system -unit unit1 -resname Appl
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

注意事項

メモリセッションフェイルオーバー用のサービスユニットを処理対象に指定した場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_unregister_application (J2EE アプリケーションの削除) (互換用)

形式

- アーカイブ形式の J2EE アプリケーションを削除する場合

```
cmx_unregister_application [共通引数] -appname <アプリケーション表示名>  
> -appver <アプリケーションバージョンID>
```

- 展開ディレクトリ形式の J2EE アプリケーションを削除する場合

```
cmx_unregister_application [共通引数] -appname <アプリケーション表示名>  
>
```

機能

Management Server が管理するドメインから、指定された J2EE アプリケーションを削除します。

- アーカイブ形式の J2EE アプリケーションを削除する場合
J2EE アプリケーションはアプリケーション表示名とアプリケーションバージョン ID で一意に識別されます。
- 展開ディレクトリ形式の J2EE アプリケーションを削除する場合
J2EE アプリケーションはアプリケーション表示名で一意に識別されます。

引数

-appname <アプリケーション表示名>

J2EE アプリケーションの表示名を指定します。cmx_register_application コマンド実行後にメッセージに出力される表示名を使用します。

-appver <アプリケーションバージョン ID>

J2EE アプリケーションのバージョン ID を指定します。

入力例

```
cmx_unregister_application -appname App1 -appver 1.0
```

戻り値

0 :
正常終了しました。

1 :
警告終了しました。

64 :
異常終了しました。

注意事項

指定した J2EE アプリケーションが cmx_define_application コマンドで Web システムに関連づけられている場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。アプリケーションの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

cmx_unregister_resource (リソースの削除) (互換用)

形式

- リソースアダプタを削除する場合
cmx_unregister_resource [共通引数] -resname <リソースアダプタ表示名>

8. Smart Composer 機能で使用するコマンド

- JavaBeans リソースを削除する場合

```
cmx_unregister_resource [共通引数] -resname <JavaBeansリソース表示名>
```

機能

Management Server が管理するドメインから、指定したリソースアダプタまたは JavaBeans リソースを削除します。

引数

-resname <リソースアダプタ表示名または JavaBeans リソース表示名>
リソースアダプタまたは JavaBeans リソースの表示名を指定します。
cmx_register_resource コマンド実行後にメッセージに出力される表示名を使用します。

入力例

```
cmx_unregister_resource -resname App1
```

戻り値

0 :
正常終了しました。

1 :
警告終了しました。

64 :
異常終了しました。

注意事項

指定されたリソースアダプタまたは JavaBeans リソースが、cmx_define_resource コマンドで Web システムに関連づけられている場合はエラーになります。

このコマンドは旧バージョンとの互換用のコマンドです。リソースの操作に関するコマンドは、08-00 以降はサーバ管理コマンドを使用することをお勧めします。

9

仮想化システムで使用する コマンド

この章では、仮想化システムで使用するコマンドの入力形式、機能などについて説明します。

9.1 仮想化システムで使用するコマンドの一覧

9.2 コマンド実行時の排他制御

9.3 仮想化システムで使用するコマンドの詳細

9.1 仮想化システムで使用するコマンドの一覧

仮想化システムで使用するコマンドの一覧を次の表に示します。

表 9-1 仮想化システムで使用するコマンドの一覧

コマンド名称	分類	概要	コマンドの系統
vmx_create_unit	管理ユニットの作成	仮想サーバマネージャに管理ユニットを新規に作成します。	更新系
vmx_delete_unit	管理ユニットの削除	仮想サーバマネージャに登録している管理ユニットのバージョン，または管理ユニット全体を削除します。	更新系
vmx_deploy_unit	管理ユニットのデプロイ	管理ユニットの仮想アプリケーションサーバのコピーをインスタンスプールに作成します。	更新系
vmx_export_rule	管理ユニット運用ルールの参照	管理ユニットで使用している管理ユニット運用ルールをエクスポートします。	参照系
vmx_get_machine_info	管理対象マシン情報の取得	すでに設定されている管理対象マシンに関する情報を取得します。	参照系
vmx_list_unit_version	管理ユニットのバージョン情報一覧表示	仮想サーバマネージャに作成している管理ユニットのバージョン情報の一覧を標準出力に表示します。	参照系
vmx_list_virtualserver_status	仮想サーバの稼働状況の一覧表示	仮想サーバのステータス一覧を標準出力に表示します。	参照系
vmx_rolling_update	管理ユニットのローリングアップデート	仮想サーバマネージャで管理している管理ユニットの，カレントバージョンに設定された仮想アプリケーションサーバを，稼働中の仮想サーバへローリングアップデートします。	更新系
vmx_scalein_unit	仮想サーバのスケールイン	指定した仮想サーバを停止します。	更新系
vmx_scaleout_unit	仮想サーバのスケールアウト	指定した管理対象マシン上に新たに一つの仮想サーバを起動します。	更新系
vmx_set_machine_info	管理対象マシン情報の設定	管理対象マシンに関する情報を設定します。	更新系
vmx_set_unit_currentversion	管理ユニットのカレントバージョンの設定	仮想サーバマネージャに登録している管理ユニットのバージョンの一つを，カレントバージョンに設定します。	更新系
vmx_start_unit	管理ユニットの一括起動	仮想サーバマネージャに登録している管理ユニットを起動します。	更新系
vmx_stop_unit	管理ユニットの一括停止	仮想サーバマネージャに登録している管理ユニットを停止します。	更新系
vmx_update_unit	管理ユニットの変更	仮想サーバマネージャに登録している管理ユニットを更新します。	更新系

(凡例)

参照系：参照系コマンド。仮想サーバマネージャから情報を取得したり、取得した情報を指定されている出力装置に表示したりするコマンドです。

更新系：更新系コマンド。仮想サーバマネージャの設定を更新したり、VMware vCenter Server にアクセスして仮想サーバを操作したりするコマンドです。

9.2 コマンド実行時の排他制御

仮想化システムで使用するコマンドおよび GUI 操作の系統ごとに、コマンド実行の排他制御の条件を次の表に示します。

表 9-2 コマンド実行の排他制御の条件

実行中のコマンド	あとから実行するコマンド		あとから実行する GUI 操作	
	参照系	更新系	参照系 ¹	更新系 ²
参照系コマンド				
更新系コマンド		×		×
参照系 GUI 操作 ¹				
更新系 GUI 操作 ²		×		×

(凡例)

：コマンドの実行が許可されます。

×：コマンドの実行を中止し、排他エラーが返ります。排他エラーとなった場合、コマンド実行ではログファイルと標準出力、GUI 操作ではログファイルと一括起動などの処理実行の失敗画面にメッセージが出力されます。

注 1

参照系の GUI 操作とは、管理ユニットビュー、管理対象マシンビュー、およびログ表示画面を表示する操作を示します。

注 2

更新系 GUI 操作とは、管理ユニットビューの一括起動ボタン、および一括停止ボタンでの操作を示します。

9.3 仮想化システムで使用するコマンドの詳細

仮想化システムで使用するコマンドの入力形式、機能などを次に示します。

コマンドの格納先

仮想化システムで使用するコマンドは、次のディレクトリに格納されています。

- Windows の場合
`<Cosminexus のインストールディレクトリ >¥manager¥vmx¥bin¥`
- Linux の場合
`/opt/Cosminexus/manager/vmx/bin/`

共通引数

各コマンドは、次の引数を共通引数として指定します。共通引数は、`vmxclient.properties` (クライアント共通設定ファイル) で、指定を省略したときのデフォルト値を設定できます。`vmxclient.properties` ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」の `vmxclient.properties` (仮想サーバマネージャのクライアント共通設定プロパティファイル) への説明を参照してください。

形式

```
コマンド名称 [-m <ホスト名>[:<ポート番号>]]
                [-u <管理ユーザID>] [-p <管理ユーザパスワード>]
```

説明

`-m <ホスト名>[:<ポート番号>]`

仮想サーバマネージャ (Management Server) のホスト名とポート番号を指定します。

- <ホスト名>
 ドット記法での IP アドレス指定 (`xxx.xxx.xxx.xxx`)、または名前解決できるホスト名で指定します。なお、`xxx` には 0 ~ 255 の整数を指定します。
- <ポート番号>
 Management Server の設定に従います。設定箇所は、`mserver.properties` ファイル (Management Server 環境設定ファイル) の `webserver.connector.http.port` キーの値となります。デフォルト値は 28080 です。

`-u <管理ユーザ ID>`

仮想サーバマネージャ (Management Server) のセットアップで設定した Management Server の管理ユーザ ID を指定します。仮想サーバマネージャのセットアップについては、マニュアル「Cosminexus アプリケーションサーバ 仮想化システム構築・運用ガイド」の仮想サーバマネージャの構築に関する説明を参照してください。

`-p <管理ユーザパスワード>`

9. 仮想化システムで使用するコマンド

仮想サーバマネージャ (Management Server) のセットアップで設定した Management Server の管理ユーザパスワードを指定します。仮想サーバマネージャのセットアップについては、マニュアル「Cosminexus アプリケーションサーバ 仮想化システム構築・運用ガイド」の仮想サーバマネージャの構築に関する説明を参照してください。

終了コード

仮想化システムで使用するコマンドは、次の範囲で戻り値を返します。

表 9-3 仮想化システムで使用するコマンドの戻り値

戻り値	意味
0	コマンドの処理が正常終了しました。
1	コマンドの処理で軽微な障害が発生しました。
2	コマンドの処理で排他エラーが発生しました。
64	コマンドの処理が異常終了しました。

注意事項

仮想化システムで使用するコマンドを実行している間は、[Ctrl] + [C] キーなどで外部から停止しないでください。

vmx_create_unit (管理ユニットの作成)

形式

```
vmx_create_unit [共通引数]
                -unit <管理ユニット名>
                -master_name <マスタ仮想サーバ名>
                -aps_dir <アプリケーションサーバ情報ディレクトリパス>
                -rule <管理ユニット運用ルールファイル>
                [-script <仮想サーバ起動時ユーザスクリプトファイル>]
```

機能

仮想サーバマネージャに管理ユニットを新規に作成します。

同じ名称の管理ユニット名を指定した場合は、作成できません。

管理ユニットの作成中に異常が発生した場合、使用できません。ただし、作成途中の管理ユニットは、VMware vCenter Server など仮想化ソフトウェアを使用して仮想サーバを起動して調査する場合に必要です。そのためバージョンとして残ります。この場合、vmx_list_unit_version コマンドの出力ヘッダ「CurrentVersion」には「x」が出力されます。

また、不要となった管理ユニットの仮想アプリケーションサーバのバージョンは vmx_delete_unit コマンドで削除できます。

引数

-unit <管理ユニット名>

新規に作成する管理ユニット名を指定します。

-master_name <マスタ仮想サーバ名>

使用するマスタ仮想サーバ名を指定します。

-aps_dir <アプリケーションサーバ情報ディレクトリパス>

アプリケーションサーバ情報ディレクトリのディレクトリパスを指定します。

-rule <管理ユニット運用ルールファイル>

管理ユニット運用ルールファイルのファイルパスを指定します。

-script <仮想サーバ起動時ユーザスクリプトパスファイル>

仮想サーバ起動時ユーザスクリプトファイルのファイルパスを指定します。

入力例

管理ユニットを管理ユニット名 (UNIT001) で新規作成する場合

```
vmx_create_unit -m HOST001 -u admin -p admin -unit UNIT001
-master_name MASTER001
-aps_dir C:¥tmp¥appDir -rule C:¥tmp¥rule.properties
-script C:¥tmp¥poweron-vm.bat
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

2:

排他エラーが発生しました。

64:

異常終了しました。

vmx_delete_unit (管理ユニットの削除)

形式

```
vmx_delete_unit [共通引数] -unit <管理ユニット名> [ -all]
```

機能

仮想サーバマネージャに登録している管理ユニットのバージョン、または管理ユニット全体を削除します。

9. 仮想化システムで使用するコマンド

-all オプションを指定しない場合は、-unit オプションで指定した管理ユニットのカレントバージョン以外の仮想アプリケーションサーバのバージョンを削除します。

-all オプションを指定すると -unit オプションで指定した管理ユニット全体を削除します。このオプションを指定する場合は、あらかじめ対象の管理ユニットに属する仮想サーバをすべて停止する必要があります。

引数

-unit <管理ユニット名>

削除対象の管理ユニット名を指定します。

-all

-unit オプションで指定した管理ユニット全体を削除する場合に指定します。

-all オプションを指定しない場合は、-unit オプションで指定した管理ユニットのカレントバージョン以外の仮想アプリケーションサーバのバージョンを削除します。

入力例

カレントバージョン以外の仮想アプリケーションサーバのバージョンを削除する場合

```
vmx_delete_unit -m HOST001 -u admin -p admin -unit UNIT001
```

管理ユニットを削除する場合

```
vmx_delete_unit -m HOST001 -u admin -p admin -unit UNIT001 -all
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

2:

排他エラーが発生しました。

64:

異常終了しました。

vmx_deploy_unit (管理ユニットのデプロイ)

形式

```
vmx_deploy_unit [共通引数] -unit <管理ユニット名>
```

機能

作成した管理ユニットの仮想アプリケーションサーバのコピーをインスタンスプールに作成し、コピーした仮想アプリケーションサーバを停止状態にします。

操作対象の管理ユニットに稼働中の仮想サーバがあった場合、稼働中の仮想サーバの処理はスキップし、そのほかの仮想サーバの処理を実行します。

管理ユニット運用ルールの設定が変更されていて、デプロイする仮想サーバの数を減らす場合に、稼働中の仮想サーバと停止されている仮想サーバがあるときは、停止されている仮想サーバを優先的に削除します。

処理の結果、仮想サーバが稼働中などの理由で、古いバージョンの仮想サーバが残ってしまった場合やデプロイ数を適切な数にできなかった場合はエラー終了します。

このコマンドがエラー終了した場合は、仮想アプリケーションサーバの状態は次のようになります。

- このコマンドがエラー終了する前に、このコマンドでデプロイされた仮想アプリケーションサーバはデプロイされた状態のままとなります。
- このコマンドがエラー終了する前に、このコマンドで削除された仮想アプリケーションサーバは削除された状態のままとなります。
- このコマンドがエラー終了する前に、このコマンドで更新された仮想アプリケーションサーバは更新された状態のままとなります。

引数

`-unit <管理ユニット名>`

デプロイする管理ユニット名を指定します。

入力例

管理ユニット (UNIT001) をデプロイする場合

```
vmx_deploy_unit -m HOST001 -u admin -p admin -unit UNIT001
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

2 :

排他エラーが発生しました。

64 :

異常終了しました。

vmx_export_rule (管理ユニット運用ルールの参照)

形式

```
vmx_export_rule [共通引数] -unit <管理ユニット名> -o <出力ファイルパス>
```

機能

管理ユニットで使用している管理ユニット運用ルールをエクスポートします。エクスポートしたファイルは、管理ユニット運用ルールのファイル形式で出力されます。

引数

-unit <管理ユニット名>

エクスポート対象の管理ユニット名を指定します。

-o <出力ファイルパス>

出力するファイルのファイルパスを指定します。

入力例

管理ユニット (UNIT001) の管理ユニット運用ルールをエクスポートする場合

```
vmx_export_rule -m HOST001 -u admin -p admin -unit UNIT001 -o  
D:¥Tmp¥rule.properties
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

vmx_get_machine_info (管理対象マシン情報の取得)

形式

```
vmx_get_machine_info [共通引数]  
-machine_info <管理対象マシン情報ファイルのパス>
```

機能

すでに設定されている管理対象マシンに関する情報を取得します。

管理対象マシン情報が設定されていない状態でこのコマンドを実行すると、エラーとなります。

引数

`-machine_info` <管理対象マシン情報ファイルのパス>

取得した管理対象マシン情報ファイルの出力先パスを指定します。

管理対象マシン情報ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」の `machine_info.xml`（管理対象マシン情報ファイル）に関する説明を参照してください。

入力例

管理対象マシン情報を取得する場合

```
vmx_get_machine_info -m HOST001 -u admin -p admin -machine_info
C:¥temp¥machine_info.xml
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

64:

異常終了しました。

vmx_list_unit_version (管理ユニットのバージョン情報一覧表示)

形式

```
vmx_list_unit_version [共通引数] [-unit <管理ユニット名>] [-h]
```

機能

仮想サーバマネージャに作成している管理ユニットのバージョン情報の一覧を標準出力に表示します。各バージョンのバージョン情報は、管理ユニットごとに降順で出力します。

`-unit` オプションを省略した場合は、すべての管理ユニットのバージョン情報一覧を管理ユニットごとに表示します。

複数の管理ユニットの情報を出力する場合は、管理ユニット名の降順で出力します。

引数

`-unit` <管理ユニット名>

対象とする管理ユニット名を指定します。

9. 仮想化システムで使用するコマンド

-h

ヘッダを出力しない場合に指定します。-h オプションを省略した場合は、ヘッダを出力します。

入力例・出力例

管理ユニット (UNIT001) のバージョン情報一覧を表示する場合

```
vmx_list_unit_version -m HOST001 -u admin -p admin -unit UNIT001
```

```
CurrentVersion,Version,UnitName,MasterName,UnitUpdateTime,MasterUpdateTime,APU
pdateTime
*,004,UNIT001,MasterVM_002,2009/08/10 15:01:23,2009/08/10 15:01:23,2009/08/10
15:01:23
,003,UNIT001,MasterVM_002,2009/08/10 13:00:00,2009/08/05 16:00:00,2009/08/10
13:00:00
```

すべての管理ユニットのバージョン情報一覧を表示する場合

```
vmx_list_unit_version -m HOST001 -u admin -p admin
```

```
CurrentVersion,Version,UnitName,MasterName,UnitUpdateTime,MasterUpdateTime,APU
pdateTime
*,001,UNIT003,MasterVM_001,2009/08/02 15:00:00,2009/08/02 15:00:00,2009/08/02
15:00:00
*,002,UNIT002,MasterVM_002,2009/08/07 11:24:11,2009/08/06 12:00:00,2009/08/07
11:24:11
,001,UNIT002,MasterVM_002,2009/08/06 12:00:00,2009/08/06 12:00:00,2009/08/06
12:00:00
*,004,UNIT001,MasterVM_002,2009/08/10 15:01:23,2009/08/10 15:01:23,2009/08/10
15:01:23
,003,UNIT001,MasterVM_002,2009/08/10 13:00:00,2009/08/05 16:00:00,2009/08/10
13:00:00
```

出力ヘッダ

CurrentVersion

カレントバージョンの場合は、アスタリスク (*) を出力します。

カレントバージョンでない場合は、半角スペースを出力します。

作成に失敗して使用できない管理ユニットの場合は、「x」を出力します。

Version

管理ユニットの仮想アプリケーションサーバのバージョンを出力します。

UnitName

管理ユニット名を出力します。

MasterName

作成元となったマスタ仮想サーバ名を出力します。

UnitUpdateTime

管理ユニットを作成または更新した日時を出力します。

MasterUpdateTime

管理ユニットの作成または更新時にマスタ仮想サーバが指定された日時を出力します。各管理ユニットで同じ日時のバージョンは、同じマスタ仮想サーバを使用していることがわかります。

APUUpdateTime

管理ユニットの作成または更新時にアプリケーションおよびアプリケーション定義情報が指定された日時を出力します。各管理ユニットで同じ日時のバージョンは、同じアプリケーションおよびアプリケーション定義情報を使用していることがわかります。

戻り値

- 0 : 正常終了しました。
- 1 : 警告終了しました。
- 64 : 異常終了しました。

vmx_list_virtualserver_status (仮想サーバの稼働状況の一覧表示)

形式

```
vmx_list_virtualserver_status [共通引数]
                               [-unit <管理ユニット名>]
                               [-hv <管理対象マシン名>]
                               [-h]
```

機能

仮想サーバのステータス一覧を標準出力に表示します。管理ユニット名および管理対象マシン名を指定することで、特定の管理ユニットおよび管理対象マシンに絞り込んだ仮想サーバのステータス一覧も表示できます。

ステータス一覧を次の表に示します。

表 9-4 ステータス一覧

ステータス	説明
setup	管理ユニットのデプロイで、仮想サーバが構築中の状態であることを示します。
starting	管理ユニットの一括起動やスケールアウトなどで、仮想サーバが起動中の状態であることを示します。

9. 仮想化システムで使用するコマンド

ステータス	説明
working	仮想サーバが起動され、仮想アプリケーションサーバが稼働している状態であることを示します。なお、障害監視機能を OFF にした場合は、仮想サーバで障害が発生しても、「working」と表示されます。
stopping	管理ユニットの一括停止やスケールインなどで、仮想サーバが停止中の状態であることを示します。
stopped	仮想サーバが停止の状態であることを示します。
delete	仮想サーバマネージャが、不要になった仮想サーバをインスタンスプールから削除している状態であることを示します。
fault(setup)	管理ユニットのデプロイで、仮想サーバが構築に失敗したことを示します。
fault(starting)	仮想サーバの起動に失敗したことを示します。
fault(working)	稼働中の仮想サーバで障害が発生したことを示します。障害監視機能を OFF にした場合は、このステータスは表示されません。
fault(stopping)	仮想サーバの停止に失敗したことを示します。
fault(delete)	仮想サーバの削除に失敗したことを示します。

注

障害監視機能については、マニュアル「Cosminexus アプリケーションサーバ 仮想化システム 構築・運用ガイド」の仮想サーバの障害監視と障害検知に関する説明を参照してください。

引数

-unit < 管理ユニット名 >

ステータス一覧表示の絞込みの対象とする管理ユニット名を指定します。

-hv < 管理対象マシン名 >

ステータス一覧表示の絞込みの対象とする管理対象マシン名を指定します。

-h

ヘッダを出力しない場合に指定します。-h オプションを省略した場合は、ヘッダを出力します。

表示形式

ヘッダに

「UnitName,UnitVersion,VirtualServerName,ManagementIP,ServiceIP,MachineName,Status」を表示し、1 行ごとに仮想サーバのステータスを CSV で表示します。

出力例

管理 LAN と業務 LAN を分けた環境の場合

```
UnitName,UnitVersion,VirtualServerName,ManagementIP,ServiceIP,MachineName,Status
UNIT1,003,VMX192168000001,192.168.0.1,192.168.2.1,HV1,working
UNIT1,003,VMX192168000002,192.168.0.2,192.168.2.2,HV1,working
```

```
UNIT2,001,VMX192168000003,192.168.0.3,192.168.2.3,HV1,starting
UNIT3,002,VMX192168000004,192.168.0.4,192.168.2.4,HV1,stopped
```

管理 LAN と業務 LAN が同じ環境の場合

```
UnitName,UnitVersion,VirtualServerName,ManagementIP,ServiceIP,MachineName,Status
UNIT1,003,VMX192168000001,192.168.0.1,192.168.0.1,HV1,working
UNIT1,003,VMX192168000002,192.168.0.2,192.168.0.2,HV1,working
UNIT2,001,VMX192168000003,192.168.0.3,192.168.0.3,HV1,starting
UNIT3,002,VMX192168000004,192.168.0.4,192.168.0.4,HV1,stopped
```

出力ヘッダ

UnitName

仮想サーバが所属する管理ユニット名を出力します。

UnitVersion

仮想サーバがデプロイされたときの管理ユニットのバージョンを出力します。

VirtualServerName

仮想サーバ名を出力します。

ManagementIP

仮想サーバの管理用ネットワークに割り当てられた IP アドレスを出力します。

ServiceIP

仮想サーバの業務用ネットワークに割り当てられた IP アドレスを出力します。

MachineName

仮想サーバがデプロイされている管理対象マシン名を出力します。

Status

仮想サーバのステータスを出力します。

注

管理 LAN と業務 LAN を同じ LAN で運用している場合は、各仮想サーバの ManagementIP および ServiceIP それぞれに同じ IP アドレスが表示されます。

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

64 :

異常終了しました。

vmx_rolling_update (管理ユニットのローリングアップデート)

形式

```
vmx_rolling_update [共通引数] -unit <管理ユニット名>
```

機能

仮想サーバマネージャで管理している管理ユニットのカレントバージョンに設定された仮想アプリケーションサーバを、稼働中の仮想サーバへローリングアップデートします。旧アプリケーションと新アプリケーションを同時に実行できる場合に使用します。ローリングアップデート実行時に、どの仮想サーバが入れ替わった（または入れ替えに失敗した）かは、仮想サーバマネージャのログに出力されます。

引数

-unit <管理ユニット名>

ローリングアップデートする管理ユニット名を指定します。

入力例

稼働中の仮想サーバへローリングアップデートする場合

```
vmx_rolling_update -m HOST001 -u admin -p admin -unit UNIT001
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

2:

排他エラーが発生しました。

64:

異常終了しました。

vmx_scalein_unit (仮想サーバのスケールイン)

形式

```
vmx_scalein_unit [共通引数] -vs <仮想サーバ名>
```

機能

指定した仮想サーバを停止します。

引数

`-vs <仮想サーバ名>`
 停止する仮想サーバ名を指定します。

戻り値

0 :
 正常終了しました。

1 :
 警告終了しました。

2 :
 排他エラーが発生しました。

64 :
 異常終了しました。

vmx_scaleout_unit (仮想サーバのスケールアウト)

形式

```
vmx_scaleout_unit [共通引数]
                  -unit <管理ユニット名> -hv <管理対象マシン名>
```

機能

指定した管理対象マシン上に新たに一つの仮想サーバを起動します。起動する仮想サーバは、コマンド実行時のカレントバージョンの管理ユニットでデプロイされたものだけが対象となります。管理対象マシンにデプロイされている仮想サーバがカレントバージョンの管理ユニットでデプロイされたものでない場合は起動対象となりません。

引数

`-unit <管理ユニット名>`
 スケールアウトする管理ユニット名を指定します。

`-hv <管理対象マシン名>`
 仮想サーバを起動する管理対象マシン名を指定します。

戻り値

0 :
 正常終了しました。

1 :
 警告終了しました。

9. 仮想化システムで使用するコマンド

2:

排他エラーが発生しました。

64:

異常終了しました。

vmx_set_machine_info (管理対象マシン情報の設定)

形式

```
vmx_set_machine_info [共通引数]
                    -machine_info <管理対象マシン情報ファイルのパス>
```

機能

管理対象マシンに関する情報を設定します。

すでに管理対象マシン情報が設定されている場合は、このコマンドで指定したファイルの情報で、設定を上書き更新します。

引数

-machine_info <管理対象マシン情報ファイルのパス>

設定する情報を記述した管理対象マシン情報ファイルのパスを指定します。

指定する管理対象マシン情報ファイルには、すべての管理対象マシンに関する情報を記述しておく必要があります。管理対象マシン情報ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編(サーバ定義)」の machine_info.xml (管理対象マシン情報ファイル) に関する説明を参照してください。

入力例

管理対象マシン情報を設定する場合

```
vmx_set_machine_info -m HOST001 -u admin -p admin -machine_info
C:¥temp¥machine_info.xml
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

2:

排他エラーが発生しました。

64:

異常終了しました。

vmx_set_unit_currentversion (管理ユニットのカルレントバージョンの設定)

形式

```
vmx_set_unit_currentversion [共通引数]
                             -unit <管理ユニット名>
                             -ver <バージョン番号>
```

機能

仮想サーバマネージャに登録している管理ユニットのバージョンの一つを、カルレントバージョンに設定します。カルレントバージョンに設定したバージョンは、管理ユニットのデプロイやスケールアウトのときの対象となります。

このコマンドは、管理ユニットが二つ以上のバージョンを持っている場合に、カルレントバージョンに設定されていないバージョンに対して実行できます。このコマンドを実行すると、カルレントバージョンだったバージョンは、カルレントバージョンではなくなり、デプロイやスケールアウトで使用されないバージョンとなります。すでにカルレントバージョンに設定されたバージョンに対してカルレントバージョンを設定すると、エラーメッセージが出力され、コマンドが終了します。

引数

-unit <管理ユニット名>

カルレントバージョンを設定する管理ユニット名を指定します。

-ver <バージョン番号>

カルレントバージョンを設定する管理ユニットのバージョン番号を指定します。

入力例

カルレントバージョンを 002 へ変更する場合

```
vmx_set_unit_currentversion -m HOST001 -u admin -p admin -unit
UNIT001 -ver 002
```

戻り値

0 :

正常終了しました。

1 :

警告終了しました。

2 :

排他エラーが発生しました。

64 :

異常終了しました。

vmx_start_unit (管理ユニットの一括起動)

形式

```
vmx_start_unit [共通引数] -unit <管理ユニット名>
```

機能

仮想サーバマネージャに登録している管理ユニットを起動します。

このコマンドでは、管理ユニット運用ルールの一括起動時に起動する仮想サーバ数 (unit.rule.start.num) だけ仮想サーバを起動します。“ unit.rule.start.num ” に指定した数の仮想サーバを起動できた場合、このコマンドは正常終了します。

起動する仮想サーバは、コマンド実行時のカレントバージョンの管理ユニットでデプロイされたものだけが対象となります。管理対象マシンにデプロイされている仮想サーバがカレントバージョンの管理ユニットでデプロイされたものでない場合は起動対象となりません。

同じ管理ユニットに属する仮想サーバがすでに一つ以上起動している場合は、次のような処理を行います。ただし、すでに起動している仮想サーバがカレントバージョンの管理ユニットでデプロイされたものでないときは、警告終了します。

- すでに起動している仮想サーバの数が “ unit.rule.start.num ” に指定した数に満たない場合は、足りない分の仮想サーバを起動します。
- すでに起動している仮想サーバの数が “ unit.rule.start.num ” に指定した数以上の場合は、起動処理は行わず、警告終了します。

仮想サーバを一つも起動できない場合はエラー終了します。起動できなかった仮想サーバが幾つかある場合は、警告終了します。管理ユニット運用ルールで、総デプロイ数が一括起動時の起動数に満たない場合は、デプロイされたすべての仮想サーバが起動できても警告終了します。起動できなかった仮想サーバは、vmx_list_virtualserver_status コマンドでは「fault(starting)」, 仮想サーバマネージャ GUI では「障害 (起動中) 」と表示されます。

引数

```
-unit <管理ユニット名>
```

起動する管理ユニット名を指定します。

入力例

管理ユニット (UNIT001) を起動する場合

```
vmx_start_unit -m HOST001 -u admin -p admin -unit UNIT001
```

戻り値

0 :

正常終了しました。

1 :
警告終了しました。

2 :
排他エラーが発生しました。

64 :
異常終了しました。

vmx_stop_unit (管理ユニットの一括停止)

形式

```
vmx_stop_unit [共通引数] -unit <管理ユニット名>
```

機能

仮想サーバマネージャに登録している管理ユニットを停止します。

管理ユニットに属する仮想サーバをすべて停止できた場合は、このコマンドは正常終了します。一部、またはすべての仮想サーバを停止できなかった場合は、このコマンドはエラー終了となります。

引数

-unit <管理ユニット名>
停止する管理ユニット名を指定します。

入力例

管理ユニット (UNIT001) を停止する場合

```
vmx_stop_unit -m HOST001 -u admin -p admin -unit UNIT001
```

戻り値

0 :
正常終了しました。

1 :
警告終了しました。

2 :
排他エラーが発生しました。

64 :
異常終了しました。

vmx_update_unit (管理ユニットの変更)

形式

主にマスタ仮想サーバを更新する場合

```
vmx_update_unit [共通引数]
                -unit <管理ユニット名>
                -master_name <マスタ仮想サーバ名>
                [-aps_dir <アプリケーションサーバ情報ディレクトリパス>]
                [-rule <管理ユニット運用ルールファイル>]
                [-script <仮想サーバ起動時ユーザスクリプトファイル>]
```

主にアプリケーションサーバ情報ディレクトリを更新する場合

```
vmx_update_unit [共通引数]
                -unit <管理ユニット名>
                -aps_dir <アプリケーションサーバ情報ディレクトリパス>
                [-master_name <マスタ仮想サーバ名>]
                [-rule <管理ユニット運用ルールファイル>]
                [-script <仮想サーバ起動時ユーザスクリプトファイル>]
```

主に管理ユニット運用ルールを更新する場合

```
vmx_update_unit [共通引数]
                -unit <管理ユニット名>
                -rule <管理ユニット運用ルールファイル>
                [-master_name <マスタ仮想サーバ名>]
                [-aps_dir <アプリケーションサーバ情報ディレクトリパス>]
                [-script <仮想サーバ起動時ユーザスクリプトファイル>]
```

主に仮想サーバ起動時ユーザスクリプトを更新する場合

```
vmx_update_unit [共通引数]
                -unit <管理ユニット名>
                -script <仮想サーバ起動時ユーザスクリプトファイル>
                [-master_name <マスタ仮想サーバ名>]
                [-aps_dir <アプリケーションサーバ情報ディレクトリパス>]
                [-rule <管理ユニット運用ルールファイル>]
```

機能

仮想サーバマネージャに登録している管理ユニットを更新します。

更新したアプリケーションやマスタ仮想サーバを、管理ユニットに反映する場合に使用します。

管理ユニットの更新中に異常が発生した場合、使用することはできませんが更新途中の管理ユニットのバージョンが残ります (VMware vCenter Server など仮想化ソフトウェアを使用して仮想サーバを起動して調査する場合に必要なため)。この場合、`vmx_list_unit_version` コマンドの出力ヘッダ「CurrentVersion」には「x」が出力されます。

また、不要となった管理ユニットの仮想アプリケーションサーバのバージョンは `vmx_delete_unit` コマンドで削除できます。

引数

`-unit <管理ユニット名>`

更新対象の管理ユニット名を指定します。

`-master_name <マスタ仮想サーバ名>`

管理ユニットに反映するマスタ仮想サーバ名を指定します。

`-aps_dir <アプリケーションサーバ情報ディレクトリパス>`

アプリケーションサーバ情報ディレクトリのディレクトリパスを指定します。

`-rule <管理ユニット運用ルールファイル>`

管理ユニット運用ルールファイルのファイルパスを指定します。

`-script <仮想サーバ起動時ユーザスクリプトファイル>`

仮想サーバ起動時ユーザスクリプトファイルのファイルパスを指定します。

入力例

更新したアプリケーションを管理ユニットに反映する場合

```
vmx_update_unit -m HOST001 -u admin -p admin -unit UNIT001
-aps_dir C:¥tmp¥appDir
```

更新したマスタ仮想サーバを管理ユニットに反映する場合

```
vmx_update_unit -m HOST001 -u admin -p admin -unit UNIT001
-master_name MASTER001
```

更新した管理ユニット運用ルールを管理ユニットに反映する場合

```
vmx_update_unit -m HOST001 -u admin -p admin -unit UNIT001 -rule
C:¥tmp¥rule.properties
```

更新した仮想サーバ起動時ユーザスクリプトを管理ユニットに反映する場合

```
vmx_update_unit -m HOST001 -u admin -p admin -unit UNIT001
-script C:¥tmp¥poweron-vm.bat
```

更新したアプリケーション, 更新したマスタ仮想サーバ, 更新した管理ユニット運用ルール, 仮想サーバ起動時ユーザスクリプトを管理ユニットに反映する場合

```
vmx_update_unit -m HOST001 -u admin -p admin -unit UNIT001
-master_name MASTER001
-aps_dir C:¥tmp¥appDir -rule C:¥tmp¥rule.properties
-script C:¥tmp¥poweron-vm.bat
```

戻り値

0:

正常終了しました。

1:

警告終了しました。

2:

9. 仮想化システムで使用するコマンド

排他エラーが発生しました。

64 :

異常終了しました。

10 JavaVM で使用するコマンド

この章では、日立の JavaVM で使用するコマンドの入力形式、機能などについて説明します。

なお、日立の JavaVM は、J2SE 5.0 および Java SE 6 に準拠しています。詳細については、マニュアル「Cosminexus アプリケーションサーバ 概説」を参照してください。また、JDK 5.0 および JDK 6 で使用できるコマンドについては、Sun Microsystems 社が提供している JDK 5.0 および JDK 6 のドキュメントを参照してください。

10.1 JavaVM で使用するコマンドの一覧

10.2 JavaVM で使用するコマンドの詳細

10.1 JavaVM で使用するコマンドの一覧

JavaVM で使用するコマンドの一覧を、次の表に示します。

表 10-1 JavaVM で使用するコマンドの一覧

コマンド名称	分類	概要
car_tar_Z	core アーカイブ機能	core ファイルと関連するライブラリなどをまとめて、一つのアーカイブファイルに出力します。このコマンドは、UNIX 用です。
eheapprof	Explicit ヒープ詳細情報付き拡張スレッドダンプの出力	Explicit ヒープ詳細情報をスレッドダンプ中に出力します。
javacore	core ファイルの取得	core ファイルとスレッドダンプを同時に取得します。このコマンドは、UNIX 用です。
javagc	ガーベージコレクションの強制発生	メモリリーク、システム障害およびアプリケーションのデバッグのために、任意のタイミングでガーベージコレクションを発生させます。
javatrace	トレース情報の収集	core ファイルからスタックトレース情報を収集します。このコマンドは、UNIX 用です。
jheapprof	日立クラス別統計情報付き拡張スレッドダンプの出力	日立クラス別統計情報をスレッドダンプ中に出力します。
jheapprofanalyzer	日立クラス別統計情報解析ファイルの CSV 出力	日立クラス別統計情報解析ファイルを CSV 形式で出力します。

注 AIX では使用できません。snapcore コマンドを使用してください。

10.2 JavaVM で使用するコマンドの詳細

JavaVM で使用するコマンドの入力形式，機能などを次に示します。

コマンドの格納先

JavaVM で使用するコマンドは，次のディレクトリに格納されています。

- Windows の場合
`<Cosminexus のインストールディレクトリ>%jdk%¥jre¥bin¥`
- UNIX の場合
`/opt/Cosminexus/jdk/jre/bin/`

car_tar_Z (core アーカイブ機能)

形式

```
car_tar_Z [-i <実行ファイル名>] [-f] [-s|-S] [<coreファイル名>] > <出力ファイル名>
```

機能

core ファイルと関連するライブラリなどをまとめて，一つのアーカイブファイルに出力します。なお，このコマンドは，UNIX 用です。

引数

-i <実行ファイル名>

実行ファイル名を指定します。

-f

パス名付きでライブラリなどのファイルを取得します。

-s

出力するアーカイブファイルのサイズ（単位：バイト）を報告します。
この場合，アーカイブファイルは出力されません。

-S

アーカイブファイルを作成するための，シェルスクリプトを出力します。
このオプションを指定する場合，<出力ファイル名>には，シェルスクリプト名を指定してください。
出力されたシェルスクリプトを実行すると，アーカイブファイルが出力されます。

<core ファイル名>

core ファイル名を指定します。省略した場合，"core" が指定されます。

<出力ファイル名>

アーカイブファイル，またはシェルスクリプトのファイル名を指定します。

入力例

アーカイブファイルを作成する場合

```
car_tar_Z -f core.8326 > corefile.tar.Z
```

アーカイブファイルのサイズを調べる場合

```
car_tar_Z -s core.8326
```

シェルスクリプトからアーカイブファイルを作成する場合

1. シェルスクリプトを作成します。

```
car_tar_Z -S core.8326 > collect_cores.csh
```

2. シェルスクリプトに実行権を付与します。

```
chmod +x ./collect_cores.csh
```

3. シェルスクリプトを実行して、アーカイブファイルを作成します。

```
./collect_cores.csh > corefile.tar.Z
```

出力メッセージ

次のメッセージを出力した場合、正常なアーカイブファイルは出力されません。

表 10-2 car_tar_Z コマンドで出力されるエラーメッセージ

項番	エラーメッセージ	説明
1	usage: car_tar_Z [-i executable-file] [-f] [-s -S] [core-file]	コマンドの引数が不正です。
2	car_tar_Z: inner error!	car_tar_Z コマンドの内部エラーです。
3	car_tar_Z: cannot create temporary name	一時的に使用するファイルが作成できません。カレントディレクトリにある、ファイル car_exec?? (?? は 00 ~ 99) を削除するか、移動してください。
4	car_tar_Z: ~ : file not found	ファイルが見つかりません。core ファイルおよび -i オプションで指定した実行ファイルを確認してください。
5	car_tar_Z: ~ : not supported platform	サポートしていない OS です。

注意事項

カレントディレクトリに、書き込み権限が必要です。

同じディレクトリ下で、同じ core ファイルまたは作成時間が同じ core ファイルに対して、car_tar_Z コマンドは同時に実行できません。

システムに tar コマンドおよび compress コマンドがインストールされている必要があります。

car_tar_Z コマンド実行中に強制終了した場合、カレントディレクトリに car_tar_Z コマンドが作成した一時的なファイルやディレクトリが残ります。

一時的に作成されるファイル：car_exec?? (?? は、00 ~ 99)

一時的に作成されるディレクトリ : `carYYMMDDhhmm/` (`YYMMDDhhmm` は、`core` ファイルの作成年月日時分)

eheapprof (Explicit ヒープ詳細情報付き拡張スレッドダンプの出力)

形式

```
eheapprof [-i|-f] [-force] -p <プロセスID>
```

機能

引数に指定したプロセス ID の java プロセスについて、Explicit ヒープ詳細情報を含んだ拡張スレッドダンプを出力します。

引数

-i

Explicit ヒープ詳細情報付き拡張スレッドダンプの出力処理の実行を確認するメッセージが表示されます。表示されたメッセージに対して `y` または `n` を入力します。このとき、`y` を入力すると、Explicit ヒープ詳細情報を含んだ拡張スレッドダンプが出力されます。`n` を入力すると、何も出力しないで処理を終了します。省略した場合、`-f` オプションが指定されないかぎり、このオプションは有効です。

-f

`-i` オプションを無効にします。省略した場合、`-i` オプションが有効になります。

-force

java プロセスが作成する `/tmp/hsperfdata_<ユーザ名>/<プロセスID>` ファイルの有無の確認をしないで、`-p` オプションで指定したプロセス ID の java プロセスに対して拡張スレッドダンプの出力を要求します。
なお、このオプションは UNIX 用のオプションです。

-p <プロセス ID>

Explicit ヒープ詳細情報を含んだ拡張スレッドダンプを出力する java プログラムのプロセス ID を指定します。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

一定時間内に Explicit ヒープ詳細情報を含んだ拡張スレッドダンプ出力処理終了の応答がありませんでした。

出力メッセージ

次のエラーメッセージまたは警告メッセージが出力された場合、Explicit ヒープ詳細情報付き拡張スレッドダンプは出力されません。

表 10-3 eheapprof コマンドで出力されるメッセージ (Windows の場合)

項番	エラーメッセージ	説明
1	usage: eheapprof [-f -i] -p process-id	eheapprof コマンドへの引数の指定が間違っています。
2	eheapprof: illegal option -- <オプション>	eheapprof コマンドに指定した <オプション> が不正です。
3	<プロセス ID>: Now processing previous request, this request canceled	eheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスが、現在 Explicit ヒープ詳細情報を出力しています。
4	<プロセス ID>: Not owner	eheapprof コマンドの引数に指定した <プロセス ID> に 0 が指定されています。
5	eheapprof: can't create work file at temporary directory, this request canceled	一時ファイル用ディレクトリに参照・書き込み権限がない場合、Explicit ヒープ詳細情報付き拡張スレッドダンプを出力できません。Explicit ヒープ詳細情報付き拡張スレッドダンプの出力要求はキャンセルされます。
6	eheapprof: can't get temporary directory, this request canceled	一時ファイル用ディレクトリが取り出せない場合、Explicit ヒープ詳細情報付き拡張スレッドダンプを出力できません。Explicit ヒープ詳細情報付き拡張スレッドダンプの出力要求はキャンセルされます。
7	eheapprof: please delete <削除できなかったファイル名> in <削除できなかったファイルのフルパス>	eheapprof コマンドを終了したときに、内部ファイルを削除できませんでした。削除できなかったファイルのフルパスにある、削除できなかったファイルを削除してください。
8	eheapprof: unexpected error occurred: <エラー原因>	eheapprof コマンド実行中に予期しないエラーが発生しました。 <エラー原因> には、例えば下記のような表示がされます。 <ul style="list-style-type: none"> • 作業用メモリ確保に失敗した場合 malloc syscall fail (errno=Y) • オブジェクトのクローズに失敗した場合 close syscall fail (errno=Y)
9	eheapprof: can't communicate with process <プロセス ID>	eheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスに問題があり、通信処理でエラーが発生しているため通信できません。または、eheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスがありません。

項番	エラーメッセージ	説明
10	<プロセス ID>: Timeout occurred. Java process not responding	eheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスから、一定時間内に Explicit ヒープ詳細情報出力処理終了の応答がありませんでした。

表 10-4 eheapprof コマンドで出力されるメッセージ (UNIX の場合)

項番	エラーメッセージ	説明
1	usage: eheapprof [-f -i] [-force] -p process-id	eheapprof コマンドへの引数の指定が間違っています。
2	eheapprof: illegal option -- <オプション>	eheapprof コマンドに指定した <オプション> が不正です。
3	<プロセス ID>: Now processing previous request, this request canceled	eheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスが、現在 Explicit ヒープ詳細情報を出力しています。
4	<プロセス ID>: No such process	eheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスがありません。または、eheapprof コマンドで指定した <プロセス ID> に該当するプロセスが java プロセス以外でした。
5	<プロセス ID>: Not owner	eheapprof コマンドの引数に指定した <プロセス ID> のプロセスのオーナーではありません。
6	eheapprof: can't create work file at /tmp, this request canceled	一時ファイル用ディレクトリに参照・書き込み権限がない場合、Explicit ヒープ詳細情報付き拡張スレッドダンプを出力できません。Explicit ヒープ詳細情報付き拡張スレッドダンプの出力要求はキャンセルされます。
7	eheapprof: please delete <削除できなかったファイル名> in <削除できなかったファイルのフルパス>	eheapprof コマンドを終了したときに、内部ファイルを削除できませんでした。削除できなかったファイルのフルパスにある、削除できなかったファイルを削除してください。
8	eheapprof: unexpected error occurred: <エラー原因>	eheapprof コマンド実行中に予期しないエラーが発生しました。 <エラー原因> には、例えば下記のような表示がされます。 <ul style="list-style-type: none"> 作業用メモリ確保に失敗した場合 malloc systemcall fail (errno=Y) オブジェクトのクローズに失敗した場合 close systemcall fail (errno=Y)
9	<プロセス ID>: Timeout occurred. Java process not responding	eheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスから、一定時間内に Explicit ヒープ詳細情報出力処理終了の応答がありませんでした。

注意事項

eheapprof コマンドはプログラムの開発用ユーティリティとして提供されているもの

です。システムの運用では使用しないでください。

同じ java プロセスに対して、同時に eheapprof コマンドは実行できません。前回の eheapprof コマンドによる Explicit ヒープ詳細情報が拡張スレッドダンプに出力されたあとに実行してください。

Windows の場合、Java プロセスは起動時に MailSlot を使った通信の初期化処理を実行します。初期化に失敗した場合、メッセージを出力して処理を中断します。

Windows の場合、引数に指定したプロセス ID の java プロセス所有者以外でも eheapprof コマンドを実行できます。

UNIX の場合、引数に指定したプロセス ID の java プロセスオーナーでないユーザがこのコマンドを実行すると、メッセージを出力して処理を終了します。ただし、ユーザが root である場合は、処理を継続します。

UNIX の場合、eheapprof コマンドは、SIGQUIT シグナルを指定されたプロセスに送信します。誤って java プログラム以外を指定すると、ほかのプログラムが停止することがあります。

UNIX の場合、eheapprof コマンド実行時に、/tmp/hsperfdata_<ユーザ名>/<プロセス ID> ファイルを使用します。該当するファイルが存在しない場合、eheapprof コマンドによる Explicit ヒープ詳細情報付き拡張スレッドダンプは出力できません。ただし、-force オプションを指定した場合、/tmp/hsperfdata_<ユーザ名>/<プロセス ID> ファイルの有無の確認をしないで、-p オプションで指定したプロセス ID の java プロセスに対して拡張スレッドダンプの出力を要求します。

javacore (core ファイルの取得)

形式

```
javacore [-i|-f] [-force] -p <プロセスID>
```

機能

コマンド実行時の core ファイルとスレッドダンプを同時に取得します。なお、このコマンドは、UNIX 用です。

引数

-i

対象プロセスのプロセス ID について、確認を要求します。省略した場合、-f オプションが指定されないかぎり、このオプションは有効です。

-f

-i オプションを無効にします。省略した場合、プロセス ID を確認するメッセージが表示されるので、y または n を入力します。このとき、y を入力すると core ファイ

ルとスレッドダンプを取得します。n を入力すると、何もしないで処理を終了します。

-force

java プロセスが作成する /tmp/hsperfdata_<ユーザ名>/<プロセス ID> ファイルの有無の確認をしないで、-p オプションで指定したプロセス ID の java プロセスに対して core を出力させます。

-p <プロセス ID>

<プロセス ID> には、core ファイルとスレッドダンプを取得する java プログラムのプロセス ID を指定します。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

一定時間内に core 生成処理終了の応答がありませんでした。

入力例

1. -f オプションを省略して javacore コマンドを実行します。

```
javacore -p 8326
```

2. プロセス ID 確認のメッセージが表示されます。

```
send SIGQUIT to 8326:?(y/n)
```

3. core ファイルおよびスレッドダンプを取得する場合は y を、取得しない場合は n を入力します。

```
send SIGQUIT to 8326:?(y/n)y
```

4. core ファイルおよびスレッドダンプを取得すると、実行中の java プログラムでは次のメッセージが出力されます。

```
Now generating core file (javacore8662.030806215140.core)...
done
```

(スレッドダンプを出力)

```
Writing Java core to javacore8662.030806215140.txt... OK
```

5. 実行中の java プログラムは、カレントディレクトリに次のファイルを作成し、プログラムを続けます。

core ファイル

10. JavaVM で使用するコマンド

```
javacore< プロセス ID>.< 日時 >.core
```

スレッドダンプ

```
javacore< プロセス ID>.< 日時 >.txt
```

出力メッセージ

次のエラーメッセージまたは警告メッセージを出力した場合、core ファイルやスレッドダンプは取得されません。

表 10-5 javacore コマンドで出力されるエラーメッセージ

項番	エラーメッセージ	説明
1	usage: javacore [-f -i] [-force] -p process-id	コマンドの引数が不正です。
2	javacore: can 't create work file at /tmp, this request canceled	/tmp に参照および書き込み権限がありません。
3	javacore: illegal option --< オプション >	コマンドの引数に指定した < オプション > が不正です。
4	javacore: unexpected error occurred:< エラー原因 >	コマンド実行中に予期しないエラーが発生しました。
5	javacore: please delete < 削除できなかったファイル名 > in < 削除できなかったファイルのフルパス >	javacore コマンドを終了したときに、内部ファイルを削除できませんでした。削除できなかったファイルのフルパスにある、削除できなかったファイルを削除してください。
6	< プロセス ID>: No such process	javacore コマンドの引数に指定した < プロセス ID > に該当するプロセスがありません。または、javacore コマンドで指定した < プロセス ID > に該当するプロセスが java プロセス以外でした。
7	< プロセス ID>: Not owner	実行ユーザは、コマンドの引数に指定した < プロセス ID > に該当するプロセスのオーナーではありません。
8	< プロセス ID>: Now processing previous request, this request canceled	コマンドの引数に指定した < プロセス ID > のプロセスが現在 core を生成中です。
9	< プロセス ID>: Timeout occurred. Java process not responding.	javacore コマンドの引数に指定した < プロセス ID > に該当するプロセスから、一定時間内に core 出力処理終了の応答がありませんでした。

注意事項

javacore コマンドは、SIGQUIT シグナルを指定されたプロセスに送信します。誤って java プログラム以外を指定すると、ほかのプログラムが停止することがあります。

同じ java プロセスに対して、同時に javacore コマンドは実行できません。前回の javacore コマンドによる core 出力処理が終了したあとに実行してください。

JavaVM プロセスが、スレッドダンプ取得要求に反応しないで無応答状態になっている場合には、javacore コマンドによる core ファイルも取得できません。この場合は、kill -6 コマンドの実行によって、JavaVM プロセスを強制停止させて core ファイルを取得してください。

AIX (5L V5.1) の場合、coredump システムコールで core を生成するため、次の環境変数を設定する必要があります。環境変数が設定されていない場合、core ファイルが正常に取得できません。

csh の場合

```
setenv LDR_CNTRL USERREGS
```

sh または ksh の場合

```
export LDR_CNTRL=USERREGS
```

LDR_CNTRL 環境変数に複数指定する場合、@ が区切り文字になります。例えば、MAXDATA と USERREGS の二つを指定する場合は、次のようになります。

csh の場合

```
setenv LDR_CNTRL MAXDATA=0x40000000@USERREGS
```

sh または ksh の場合

```
export LDR_CNTRL=MAXDATA=0x40000000@USERREGS
```

Linux の場合、gdb の gcore コマンドで core を生成します。gdb がインストールされていない場合には、次のエラーメッセージが標準出力に出力されます。

```
Error occurred in generating core file, gdb not found.
```

また、インストールされている gdb のバージョンが古い場合には、次のエラーメッセージが標準出力に出力されます。

```
Error occurred in generating core file, gdb version 5.2 or later needed.
```

javacore コマンドは、実行時に、/tmp/hisperfdata_<ユーザ名>/<プロセス ID> ファイルを使用します。該当するファイルが存在しない場合は、javacore コマンドによる core ファイルの出力はできません。ただし、-force オプションを指定した場合、/tmp/hisperfdata_<ユーザ名>/<プロセス ID> ファイルの有無の確認をしないで、-p オプションで指定したプロセス ID の java プロセスに対して core を出力させます。

javagc (ガーベージコレクションの強制発生)

形式

```
javagc [-i|-f] [-v] [-s] [-force] -p <プロセスID>
```

機能

メモリリーク、システム障害およびアプリケーションのデバッグのために、プロセス ID

が <プロセス ID> の java プロセスに対して、任意のタイミングでガーベージコレクションを発生させます。

java プロセスとの通信には SIGQUIT シグナルを使用します。コマンドを実行すると、ガーベージコレクションを発生させるかどうか (Windows の場合)、またはガーベージコレクションを発生させるために SIGQUIT シグナルを送信するかどうか (UNIX の場合) をユーザに確認します。発生させない、または送信しない (n) と回答した場合には、ガーベージコレクションは発生しません (戻り値が 1 となります)。この確認動作は、-f オプションを指定することで省略できます。

<プロセス ID> の java プロセスが通常の要因で発生するコピーガーベージコレクションやフルガーベージコレクションを実行中の場合は、その終了を待ってからコマンドを実行します。

引数

-i

<プロセス ID> のプロセスに対して、ガーベージコレクションを発生させるかどうか (Windows の場合)、または SIGQUIT シグナルを送信させるかどうか (UNIX の場合) をユーザに確認します。
このオプションより前に指定した、-f オプションを無効にします。

-f

<プロセス ID> のプロセスに対して、ガーベージコレクションを発生させるかどうか (Windows の場合)、または SIGQUIT シグナルを送信させるかどうか (UNIX の場合) をユーザに確認しません。
このオプションより前に指定した、-i オプションを無効にします。

-v

- -XX:+HitachiVerboseGC オプションが指定されている場合に出力される、日立 JavaVM ログファイルと同じ内容を、日立 JavaVM ログファイルに出力します。
- -XX:+HitachiVerboseGC オプションが指定されている場合、その日立 JavaVM ログファイルに情報を追加します。
- -XX:+HitachiVerboseGCPrintDate, -XX:+HitachiVerboseGCIntervalTime, -XX:+HitachiVerboseGCPrintCause, -XX:+HitachiVerboseGCCpuTime, -XX:+HitachiCommaVerboseGC などの追加オプションが指定されている場合は、その形式に従って出力されます。
- -XX:HitachiVerboseGCIntervalTime オプションが指定され、指定時間を経過していない場合でも日立 JavaVM ログファイルを出力します。
- -XX:+HitachiVerboseGC オプションが指定されていない場合は、日立 JavaVM ログファイル作成の規則に従って新規に日立 JavaVM ログファイルを作成します。

-s

- -XX:+HitachiVerboseGC オプションが指定されている場合に出力される、日立

JavaVM ログファイルと同じ内容を、標準出力に出力します。

- `-XX:+HitachiVerboseGCPrintDate` , `-XX:+HitachiVerboseGCIntervalTime` ,
`-XX:+HitachiVerboseGCPrintCause` , `-XX:+HitachiCommaVerboseGC` などの追加オプションが指定されている場合は、その形式に従って出力されます。
- `-XX:HitachiVerboseGCIntervalTime` オプションが指定され、指定時間を経過していない場合でも日立 JavaVM ログファイルと同じ内容を標準出力に出力します。

`-force`

java プロセスが作成する `/tmp/hsperfdata_<ユーザ名>/<プロセス ID>` ファイルの有無の確認をしないで、`-p` オプションで指定したプロセス ID の java プロセスに対してガーベージコレクションを発生させます。
なお、このオプションは UNIX 用のオプションです。

`-p <プロセス ID>`

プロセス ID を指定します。

日立固有の JavaVM 拡張オプションについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」を参照してください。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

一定時間内にガーベージコレクション処理終了の応答がありませんでした。

入力例

1. `-i` オプションを指定して `javagc` コマンドを実行します。

```
javagc -i -v -p 8326
```

2. Windows の場合、ガーベージコレクションを発生させるかどうかの確認メッセージが表示されます。UNIX の場合、プロセス ID の確認メッセージが表示されます。

- Windows の場合

```
Force VM to execute GC ? (y/n)
```

- UNIX の場合

```
send SIGQUIT to 8326:?(y/n)
```

3. Windows の場合、ガーベージコレクションを発生させる場合は `y` を、発生させない場合は `n` を入力します。UNIX の場合、SIGQUIT シグナルを送信させる場合は `y` を、送信させない場合は `n` を入力します。

10. JavaVM で使用するコマンド

- Windows の場合
Force VM to execute GC ? (y/n)y
- UNIX の場合
send SIGQUIT to 8326:?(y/n)y

出力例

```
[VGC]<Wed Mar 17 00:42:30 2004>(Skip Full:0, Copy:0) [Full GC 149K->149K(1984K),  
0.0786038 secs] [DefNew::Eden: 264K->0K(512K)] [DefNew::Survivor:  
0K->63K(64K)] [Tenured: 85K->149K(1408K)] [Perm:  
1300K->1300K(1536K)] [cause:JavaGC Command]
```

出力メッセージ

表 10-6 javagc コマンドで出力されるメッセージ (Windows の場合)

項番	メッセージ	説明
1	usage: javagc [-f -i] [-v] [-s] -p process-id	javagc コマンドへの引数の指定が間違っています。
2	javagc: illegal option--< オプション >	javagc コマンドに指定した < オプション > が不正です。
3	javagc: can't communicate with process < プロセス ID>	javagc コマンドの引数に指定した < プロセス ID > に問題があり、通信処理でエラーが発生しているため通信できません。または、javagc コマンドの引数に指定した < プロセス ID > に該当するプロセスがありません。
4	< プロセス ID >: Not owner	javagc コマンドで指定した < プロセス ID > に 0 が指定されています。
5	< プロセス ID >: Now processing previous request, this request canceled	javagc コマンドの引数に指定した < プロセス ID > に該当するプロセスは、前回の javagc コマンドによるガーベージコレクションを実行中です。javagc コマンドによるガーベージコレクション実行要求はキャンセルされます。
6	javagc: can ' t create work file at temporary directory, this request canceled	一時ファイル用ディレクトリに参照・書き込み権限がない場合、ガーベージコレクション要求ファイルが作成できません。このガーベージコレクション実行要求はキャンセルされます。
7	javagc: can't get temporary directory, this request canceled	一時ファイル用ディレクトリが取り出せない場合、ガーベージコレクション要求ファイルが作成できません。このガーベージコレクション実行要求はキャンセルされます。
8	javagc: unexpected error occurred:< エラー原因 >	javagc コマンド実行中に予期しないエラーが発生しました。 < エラー原因 > には、例えば下記のような表示がされます。 <ul style="list-style-type: none">• 作業用メモリ確保に失敗した場合 malloc systemcall fail (errno=Y)• オブジェクトのクローズに失敗した場合 close systemcall fail (errno=Y)

項番	メッセージ	説明
9	<プロセス ID>: Timeout occurred. Java process not responding.	javagc コマンドの引数に指定した <プロセス ID> に該当するプロセスから、一定時間内にガーベージコレクション処理終了の応答がありませんでした。
10	javagc : please delete <削除できなかったファイル名 >in <削除できなかったファイルのフルパス >	javagc コマンドを終了したときに、内部ファイルを削除できませんでした。削除できなかったファイルのフルパスにある、削除できなかったファイルを削除してください。

表 10-7 javagc コマンドで出力されるメッセージ (UNIX の場合)

項番	メッセージ	説明
1	javagc [-f -i][-v][-s] [-force] -p process-id	javagc コマンドへの引数の指定が間違っています。
2	javagc: illegal option<< オプション >>	javagc コマンドに指定した <オプション > が不正です。
3	<プロセス ID>: No such process	javagc コマンドの引数に指定した <プロセス ID> に該当するプロセスがありません。または、javagc コマンドで指定した <プロセス ID> に該当するプロセスが java プロセス以外でした。
4	<プロセス ID>: Not owner	実行ユーザは、javagc コマンドの引数に指定した <プロセス ID> に該当するプロセスのオーナーではありません。
5	<プロセス ID>: Now processing previous request, this request canceled	javagc コマンドの引数に指定した <プロセス ID> に該当するプロセスは、前回の javagc コマンドによるガーベージコレクションを実行中です。javagc コマンドによるガーベージコレクション実行要求はキャンセルされます。
6	javagc: can't create work file at / tmp, this request canceled	/tmp に参照および書き込み権限がないため、ガーベージコレクション要求ファイルが作成できません。javagc コマンドによるガーベージコレクション実行要求はキャンセルされます。
7	javagc: unexpected error occurred:< エラー原因 >	javagc コマンド実行中に予期しないエラーが発生しました。 <エラー原因 > には、例えば下記のような表示がされます。 <ul style="list-style-type: none"> • 作業用メモリ確保に失敗した場合 malloc systemcall fail (errno=Y) • オブジェクトのクローズに失敗した場合 close systemcall fail (errno=Y)
8	<プロセス ID>: Timeout occurred. Java process not responding.	javagc コマンドの引数に指定した <プロセス ID> に該当するプロセスから、一定時間内にガーベージコレクション処理終了の応答がありませんでした。

10. JavaVM で使用するコマンド

項番	メッセージ	説明
9	javagc : please delete <削除できなかったファイル名 >in <削除できなかったファイルのフルパス >	javagc コマンドを終了したときに、内部ファイルを削除できませんでした。削除できなかったファイルのフルパスにある、削除できなかったファイルを削除してください。

注意事項

同じ java プロセスに対して、同時に javagc コマンドは実行できません。前回の javagc コマンドによるガーベージコレクション処理が終了してから実行してください。前回のガーベージコレクション処理が終了している場合は、日立 JavaVM 固有ログファイルに出力される拡張 verbosegc 機能のガーベージコレクションの要因に "JavaGC Command" が出力されます。

Windows の場合、引数に指定したプロセス ID の java プロセス所有者以外でも javagc コマンドを実行できます。

UNIX の場合、javagc コマンドは SIGQUIT シグナルを指定されたプロセス ID に送信します。誤って java プロセス以外のプロセス ID を指定すると、ほかのプログラムが停止することがあります。

UNIX の場合、javagc コマンド実行時に、/tmp/hsperfdata_<ユーザ名>/<プロセス ID> ファイルを使用します。該当するファイルが存在しない場合は、javagc コマンドによるガーベージコレクションの要求はできません。ただし、-force オプションを指定した場合、/tmp/hsperfdata_<ユーザ名>/<プロセス ID> ファイルの有無の確認をしないで、-p オプションで指定したプロセス ID の java プロセスに対してガーベージコレクションを発生させます。

javatrace (トレース情報の収集)

形式

```
javatrace <coreファイル名> <実行ファイル名> [<出力ファイル名>] [-l <ライブラリファイル名> ...]
```

機能

core ファイルからスタックトレース情報を取得します。なお、このコマンドは、UNIX 用です。

このコマンドは、JavaVM が異常終了して core ファイルを生成した場合、その異常終了の原因究明に必要な情報を取得するために実行するコマンドです。プロセスダウン時の詳細要因などが調査できます。

引数

<core ファイル名>

core ファイル名を指定します。

< 実行ファイル名 >

core ファイルを生成した実行ファイル名を指定します。

< 出力ファイル名 >

出力ファイル名を指定します。

省略した場合は、カレントディレクトリの "javatrace.log" に出力されます。

-l < ライブラリファイル名 >

使用したライブラリファイルを指定します。

共用ライブラリが絶対パスで実行ファイルに取り込まれている場合は、自動的に読み込まれるため、指定する必要はありません。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

入力例

JavaVM が異常終了して core ファイルを生成した場合、次のメッセージが表示されます。このメッセージ内の javatrace コマンドの文字列を実行します。

なお、このメッセージは、異常終了時に生成されるエラーレポートファイル (hs_err_pid< プロセス ID>.log) にも出力されます。

```

:
# You can get further information from javatrace.log file generated
# by using javatrace command.
# usage: javatrace core-file-name loadmodule-name [out-file-name]
# [-l(library-name)...]
# Please use javatrace command as follows and submit a bug report
# to Hitachi with javatrace.log file:
# [/opt/Cosminexus/jdk/jre/bin/javatrace core /opt/Cosminexus/CC/server/bin/
# cjstartsv]
#

```

core ファイル名が core の場合

```

/opt/Cosminexus/jdk/jre/bin/javatrace core /opt/Cosminexus/CC/
server/bin/cjstartsv

```

OSによっては実際に出力される core ファイル名が core.<プロセス ID> になる場合があります。その場合は、実際に出力された core ファイル名を javatrace の引数に指定してください。

core ファイル名が core.<プロセス ID> の場合

10. JVM で使用するコマンド

```
/opt/Cosminexus/jdk/jre/bin/javatrace core.8326 /opt/  
Cosminexus/CC/server/bin/cjstartsv
```

出力メッセージ

次のエラーメッセージまたは警告メッセージを出力した場合、スタックトレース情報は出力されません。

表 10-8 javatrace コマンドで出力されるエラーメッセージ

項番	エラーメッセージ	説明
1	usage : javatrace core-filename loadmodule-name [out-filename] [-library-name...]	javatrace コマンドへの引数の指定が間違っています。
2	javatrace: Cannot open file-name :No such file or directory	<出力ファイル> で指定したファイル以外の ファイル (file-name) が見つかりません。
3	javatrace: Cannot open file-name : Permission denied	ファイル (file-name) の読み込みが許可されて いません。
4	javatrace : Cannot create file-name : Already exist	<出力ファイル> に指定したファイル (file-name) がすでに存在しています。
5	* unknown core type(XXXXXXX) ignored.	<core ファイル名> で指定したファイルは、 core ファイルではありません。
6	* ERROR : file-name, unknown magic(0xXXXX)	<実行ファイル名> で指定したファイルは、 実行ファイルではありません。
7	WARNING : core file may not match loadmodule (core file from ' loadmodule-name')	<実行ファイル名> で指定した実行ファイル と、<core ファイル名> で指定した core ファ イルから取り出した実行ファイル (loadmodule-name) が一致していません。
8	javatrace : Bad argument : argument-name is directory	指定したファイル (file-name) はディレクト リです。
9	* ERROR : library-name open : No such file or directory	実行ファイルが取り込んだ共用ライブラリ (library-name) が見つかりません。共用ライ ブラリが相対パスで取り込まれている場合 には、-l オプションで共用ライブラリを明示 的に指定する必要があります。
10	javatrace : illegal option -- x	誤ったオプション文字 (x) が指定されていま す。
11	javatrace : Cannot create file-name : Permission denied	<出力ファイル> で指定したファイル (file-name) の出力先に書き込み権限があり ません。

注意事項

javatrace コマンドは core ファイルが生成されたマシンで実行してください。

jheapprof (日立クラス別統計情報付き拡張スレッドダンプの出力)

形式

```
jheapprof [-i|-f] [-class <クラス名>] [-explicit|-noexplicit]
          [-fullgc|-copygc|-nogc] [-garbage|-nogarbage]
          [-rootobjectinfo|-norootobjectinfo]
          [-rootobjectinfost <値> ] [-force] -p <プロセスID>
```

機能

引数に指定したプロセス ID の java プロセスについて、日立クラス別統計情報を含んだ拡張スレッドダンプを出力します。

引数

-i

日立クラス別統計情報付き拡張スレッドダンプの出力処理の実行を確認するメッセージが表示されます。表示されたメッセージに対して y または n を入力します。このとき、y を入力すると、日立クラス別統計情報を含んだ拡張スレッドダンプが出力されます。n を入力すると、何も出力しないで処理を終了します。省略した場合、-f オプションが指定されないかぎり、このオプションは有効です。

-f

-i オプションを無効にします。省略した場合、-i オプションが有効になります。

-class <クラス名>

<クラス名> に指定したクラス (インスタンス) をメンバに持つクラスの構造を一覧にしてスレッドダンプ中に出力します。

-explicit

インスタンス統計機能の統計対象に Explicit ヒープを含めます。なお、このオプションと -noexplicit オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-noexplicit

インスタンス統計機能の統計対象に Explicit ヒープを含めません。省略した場合、-explicit オプションが有効になります。なお、このオプションと -explicit オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-fullgc

統計する前に実行するガーベージコレクションにフルガーベージコレクションを設定します。なお、このオプションと -copygc オプションまたは -nogc オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

10. JavaVM で使用するコマンド

-copygc

統計する前に実行するガーベージコレクションにコピーガーベージコレクションを設定します。省略した場合、-fullgc オプションが有効になります。

なお、このオプションと -fullgc オプションまたは -nogc オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-nogc

統計する前にガーベージコレクションを実行しません。省略した場合、-fullgc オプションが有効になります。なお、このオプションと -fullgc オプションまたは -copygc オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-garbage

Tenured 領域内不要オブジェクト統計機能が有効になり、Tenured 領域内の不要なオブジェクトを統計対象とした日立クラス別統計情報を出力します。また、インスタンス統計機能と STATIC メンバ統計機能は無効になります。省略した場合、-nogarbage オプションが有効になります。統計前ガーベージコレクション選択機能については、-fullgc オプション、-copygc オプションが無効になり、-nogc オプションは有効になります。このため、統計処理前にガーベージコレクションを実行しません。なお、このオプションと -nogarbage オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-nogarbage

Tenured 領域内不要オブジェクト統計機能が無効になります。そのため、Tenured 領域内の不要なオブジェクトを統計対象とした日立クラス別統計情報は出力しません。なお、このオプションと -garbage オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-rootobjectinfo

Tenured 増加要因の基点オブジェクトリスト出力機能が有効になり、Tenured 増加要因の基点オブジェクトリストを出力します。

このオプションは、-garbage オプションが有効であることが前提です。そのため、-nogarbage オプションを有効にすると、このオプションは無効になります。なお、このオプションと -norootobjectinfo オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-norootobjectinfo

Tenured 増加要因の基点オブジェクトリスト出力機能が無効になります。そのため、Tenured 増加要因の基点オブジェクトリストは出力しません。省略した場合、-rootobjectinfo オプションが有効になります。なお、このオプションと -rootobjectinfo オプションを同時に指定している場合、最後に指定しているオプションが有効になります。

-rootobjectinfo < 値 >

Tenured 増加要因の基点オブジェクトリストの情報量を調節します。インスタンスサイズの合計が、指定した < 値 > 以上のクラス情報だけが、Tenured 増加要因の基点オブジェクトリストに出力されます。省略した場合、0 が設定されます。

このオプションは、-rootobjectinfo オプションが有効であることが前提です。そのため、-norootobjectinfo オプションを有効にすると、このオプションは無効になります。< 値 > には 0 以上の自然数を指定できます。自然数以外や、文字列を指定した場合は、引数の指定に誤りがあるという内容のエラーメッセージが出力されて終了します。

-force

java プロセスが作成する /tmp/hsperfdata_<ユーザ名>/<プロセス ID> ファイルの有無の確認をしないで、-p オプションで指定したプロセス ID の java プロセスに対して拡張スレッドダンプの出力を要求します。

なお、このオプションは UNIX 用のオプションです。

-p < プロセス ID >

< プロセス ID > には、日立クラス別統計情報を出力する java プログラムのプロセス ID を指定します。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

一定時間内に日立クラス別統計情報出力処理終了の応答がありませんでした。

入力例

1. -f オプションを省略して jheapprof コマンドを実行します。

```
% jheapprof -p 2463
```

2. Windows の場合、日立クラス別統計情報付き拡張スレッドダンプを出力するかどうかの確認メッセージが表示されます。UNIX の場合、プロセス ID の確認メッセージが表示されます。

- Windows の場合

```
Force VM to output HitachiJavaHeapProfile: ? (y/n)
```

- UNIX の場合

```
send SIGQUIT to 2463: ? (y/n)
```

3. Windows の場合、日立クラス別統計情報付き拡張スレッドダンプを出力する場合は y

10. JavaVM で使用するコマンド

を、出力しない場合は `n` を入力します。UNIX の場合、SIGQUIT シグナルを送信させる場合は `y` を、送信させない場合は `n` を入力します。

- Windows の場合

```
Force VM to output HitachiJavaHeapProfile: ? (y/n)y
```

- UNIX の場合

```
send SIGQUIT to 2463: ? (y/n)y
```

4. 日立クラス別統計情報付き拡張スレッドダンプを出力すると、実行中の java プログラムでは次のメッセージが出力されます。

```
Writing Java core to javacore2463.030806215140.txt... OK
```

5. 実行中の java プログラムは、カレントディレクトリに日立クラス別統計情報付き拡張スレッドダンプ (`javacore< プロセス ID>.< 日時 >.txt`) を作成し、プログラムを継続します。

出力形式

日立クラス別統計情報の出力形式については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」のクラス別統計情報機能の説明を参照してください。

出力メッセージ

次のエラーメッセージまたは警告メッセージが出力された場合、日立クラス別統計情報付き拡張スレッドダンプは出力されません。

表 10-9 jheapprof コマンドで出力されるメッセージ (Windows の場合)

項番	エラーメッセージ	説明
1	usage: jheapprof [-f -i] [-class classname] [-explicit -noexplicit] [-fullgc -copygc -noge] [-garbage -nogarbage] [-rootobjectinfo -norootobjectinfo] [-rootobjectinfofst size] -p process-id	jheapprof コマンドへの引数の指定が間違っています。
2	jheapprof: illegal option -- <オプション>	jheapprof コマンドに指定した <オプション> が不正です。
3	<プロセス ID>: Now processing previous request, this request canceled	jheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスが、クラス別統計情報の出力中です。
4	<プロセス ID>: Not owner	jheapprof コマンドの引数に指定した <プロセス ID> に 0 が指定されています。

項番	エラーメッセージ	説明
5	jheapprof: can't create work file at temporary directory , this request canceled	一時ファイル用ディレクトリに参照・書き込み権限がない場合、日立クラス別統計情報付き拡張スレッドダンプを出力できません。日立クラス別統計情報付き拡張スレッドダンプの出力要求はキャンセルされます。
6	jheapprof: can't get temporary directory, this request canceled	一時ファイル用ディレクトリが取り出せない場合、日立クラス別統計情報付き拡張スレッドダンプを出力できません。日立クラス別統計情報付き拡張スレッドダンプの出力要求はキャンセルされます。
7	jheapprof: please delete <削除できなかったファイル名 > in <削除できなかったファイルのフルパス >	jheapprof コマンドを終了したときに、内部ファイルを削除できませんでした。削除できなかったファイルのフルパスにある、削除できなかったファイルを削除してください。
8	jheapprof: unexpected error occurred: <エラー原因 >	jheapprof コマンド実行中に予期しないエラーが発生しました。 <エラー原因 > には、例えば下記のような表示がされます。 <ul style="list-style-type: none"> 作業用メモリ確保に失敗した場合 malloc syscall fail (errno=Y) オブジェクトのクローズに失敗した場合 close syscall fail (errno=Y)
9	jheapprof: can't communicate with process <プロセス ID>	jheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスに問題があり、通信処理でエラーが発生しているため通信できません。または、jheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスがありません。
10	<プロセス ID>: Timeout occurred. Java process not responding	jheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスから、一定時間内に日立クラス別統計出力処理終了の応答がありませんでした。

表 10-10 jheapprof コマンドで出力されるメッセージ (UNIX の場合)

項番	エラーメッセージ	説明
1	usage: jheapprof [-f -i] [-class classname] [-explicit -noexplicit] [-fullgc -copygc -nogc] [-garbage -nogarbage] [-rootobjectinfo -norootobjectinfo] [-rootobjectinfo size] [-force] -p process-id	jheapprof コマンドへの引数の指定が間違っています。
2	jheapprof: illegal option -- <オプション >	jheapprof コマンドに指定した <オプション > が不正です。
3	<プロセス ID>: Now processing previous request, this request canceled	jheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスが現在クラス別統計情報の出力中です。

10. JavaVM で使用するコマンド

項番	エラーメッセージ	説明
4	<プロセス ID>: No such process	jheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスがありません。または、jheapprof コマンドで指定した <プロセス ID> に該当するプロセスが java プロセス以外でした。
5	<プロセス ID>: Not owner	jheapprof コマンドの引数に指定した <プロセス ID> のプロセスのオーナーではありません。
6	jheapprof: can't create work file at /tmp, this request canceled	一時ファイル用ディレクトリに参照・書き込み権限がない場合、日立クラス別統計情報付き拡張スレッドダンプを出力できません。日立クラス別統計情報付き拡張スレッドダンプの出力要求はキャンセルされます。
7	jheapprof: please delete <削除できなかったファイル名 >in <削除できなかったファイルのフルパス >	jheapprof コマンドを終了したときに、内部ファイルを削除できませんでした。削除できなかったファイルのフルパスにある、削除できなかったファイルを削除してください。
8	jheapprof: unexpected error occurred: <エラー原因 >	jheapprof コマンド実行中に予期しないエラーが発生しました。<エラー原因 >には、例えば下記のような表示がされます。 <ul style="list-style-type: none"> 作業用メモリ確保に失敗した場合 malloc systemcall fail (errno=Y) オブジェクトのクローズに失敗した場合 close systemcall fail (errno=Y)
9	<プロセス ID>: Timeout occurred. Java process not responding	jheapprof コマンドの引数に指定した <プロセス ID> に該当するプロセスから、一定時間内に日立クラス別統計情報出力処理終了の応答がありませんでした。

注意事項

jheapprof コマンドはプログラムの開発用ユーティリティとして提供されているものです。システムの運用では使用しないでください。

同じ java プロセスに対して、同時に jheapprof コマンドは実行できません。前回の jheapprof コマンドによるクラス別統計情報が拡張スレッドダンプに出力されたあとに実行してください。

Windows の場合、Java プロセスは起動時に MailSlot を使った通信の初期化処理を実行します。初期化に失敗した場合、メッセージを出力して処理を中断します。

Windows の場合、引数に指定したプロセス ID の java プロセス所有者以外でも jheapprof コマンドを実行できます。

UNIX の場合、引数に指定したプロセス ID の java プロセスオーナーでないユーザがこのコマンドを実行すると、メッセージが出力されて、処理は終了します。ただし、ユーザが root である場合は、処理を継続します。

UNIX の場合、jheapprof コマンドは、SIGQUIT シグナルを指定されたプロセスに送

信します。誤って java プログラム以外を指定すると、ほかのプログラムが停止することがあります。

UNIX の場合、jheapprof コマンド実行時に、/tmp/hsperfdata_<ユーザ名>/<プロセス ID> ファイルを使用します。該当するファイルが存在しない場合、jheapprof コマンドによるクラス別統計情報出力はできません。

Explicit ヒープをインスタンス統計情報の統計対象に含めるためには、-XX:+HitachiUseExplicitMemory を指定して、明示管理ヒープ機能を有効にしている java プロセスに対して、jheapprof コマンドで -explicit オプションを指定してください。なお、-XX:-HitachiUseExplicitMemory を指定して、明示管理ヒープ機能を無効にしている場合、-explicit を指定しても Explicit ヒープ内のインスタンスはインスタンス統計情報の統計対象になりません。

jheapprofanalyzer (日立クラス別統計情報解析ファイルの CSV 出力)

形式

```
jheapprofanalyzer [-J <オプション名>] [<ファイル名>]
```

機能

日立クラス別統計情報解析ファイルを CSV 形式で出力します。

日立クラス別統計情報解析ファイルで使用する入力ファイルについては、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」のクラス別統計情報解析機能の説明を参照してください。

引数

-J <オプション名>

<オプション名> には、次のオプションを指定できます。また、次のオプション以外を指定した場合は、動作保証の対象外となります。

- -Xms
メモリ割り当てプールの初期サイズをバイト数で指定します。
- -Xmx
メモリ割り当てプールの最大サイズをバイト数で指定します。
- -DJP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold=num
num: インスタンス合計サイズのしきい値を設定します。範囲は 0 ~ 2⁶³-1 (Long.MAX_VALUE) です。インスタンス合計サイズが num 以上のクラスだけ出力します。デフォルト値は、1024 です。

<ファイル名>

日立クラス別統計情報付き拡張スレッドダンプファイルを指定できます。ファイル

10. JavaVM で使用するコマンド

名称の規定は、特にありません。また、ファイルの指定は、順不同であり、数に制限はありません。

戻り値

0:

正常終了しました。

1 以上:

異常終了しました。

入力例

```
jheapprofanalyzer -J-Xms1024m -J-Xmx1024m  
-J-DJP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold=5000  
javacore22356.080523161703.txt javacore22356.080523161711.txt
```

出力形式

日立クラス別統計情報解析ファイルの出力形式については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」のクラス別統計情報解析機能の説明を参照してください。

出力メッセージ

次のエラーメッセージが出力された場合、日立クラス別統計情報解析ファイルは出力されません。また、次のエラーメッセージ以外が出力された場合は、デフォルトの例外処理となります。

表 10-11 jheapprofanalyzer コマンドで出力されるエラーメッセージ

項番	エラーメッセージ	説明	出力後の動作
1	usage: jheapprofanalyzer [options] file... where options include: -J-Xms<size> set initial Java heap size -J-Xmx<size> set maximum Java heap size -J-DJP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold=<num> set instance total size threshold	JheapprofAnalyzer クラスへの引数の指定が間違っています。	(a)
2	JheapprofAnalyzer: Illegal property value<num>. Default is assumed.	JP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold の <num> に数字以外を指定しました。または <num> が範囲外となっています。	(b)
3	JheapprofAnalyzer: can't open input file<ファイル名>	ディレクトリに <ファイル名> がありません。または、別の原因でファイルを開けません。	(c)

項番	エラーメッセージ	説明	出力後の動作
4	JheapprofAnalyzer: can't read input file<ファイル名>	<ファイル名>の読み込みに失敗しました。	(c)
5	JheapprofAnalyzer: Illegal input file format<ファイル名>	<ファイル名>は、日立クラス別統計情報付き拡張スレッドダンプファイルではありません。	(c)
6	JheapprofAnalyzer: can't open output file<ファイル名>	出力ファイルが開けません。 エラーの原因として、次の状態が考えられます。 <ul style="list-style-type: none"> • 出力ファイルがディレクトリとなっています。 • 出力ファイルがありません。 • 別の原因で出力ファイルを開けません。 	(a)
7	JheapprofAnalyzer: can't write output file<ファイル名>	<ファイル名>の書き込みに失敗しました。	(a)

(凡例)

- (a) : エラーとなり処理を終了します。
- (b) : デフォルトを仮定して処理を続行します。
- (c) : 処理を続行し、指定したすべての入力ファイルのエラーチェックをします。

注意事項

日立クラス別統計情報解析機能では、日付を取得するときと、データを読み込むときにファイルを開きます。そのため、コマンド実行中に入力ファイルの更新および削除の操作をした場合の結果は保証されません。

11 ログの運用で使用するコマンド

この章では、ログの運用で使用するコマンドの入力形式、機能などについて説明します。

11.1 ログの運用で使用するコマンドの一覧

11.2 ログの運用で使用するコマンドの詳細

11.1 ログの運用で使用するコマンドの一覧

ログの運用で使用するコマンドの一覧を、次の表に示します。

表 11-1 ログの運用で使用するコマンドの一覧

コマンド名称	分類	概要
cjldellog	Java アプリケーションのシステムログ出力先サブディレクトリの削除	Java アプリケーションのシステムログ出力先サブディレクトリを削除します。 cjldellog コマンドについては、「14. Java アプリケーションで使用するコマンド」の「cjldellog」を参照してください。
snapshotlog	snapshot ログの収集	Cosminexus システムの構成ソフトウェアが出力したログを収集します。

11.2 ログの運用で使用するコマンドの詳細

ログの運用で使用する各コマンドの入力形式、機能などを次に示します。

コマンドの格納先

ログの運用で使用するコマンドは、次のディレクトリに格納されています。

- Windows の場合
`<Cosminexus のインストールディレクトリ>%manager%bin%`
- UNIX の場合
`/opt/Cosminexus/manager/bin/`

snapshotlog (snapshot ログの収集)

形式

`snapshotlog <出力ファイル名> [<snapshot ログ収集対象定義ファイル名>]`

機能

snapshot ログを収集します。snapshot ログは、業務システムにトラブルが発生した場合に、調査用資料として利用できます。このコマンドは、Management Server を利用しないで運用している場合も実行できます。

snapshot ログ情報については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

引数

< 出力ファイル名 >

収集した snapshot ログを出力する ZIP ファイルを指定します。

ファイル名だけを指定した場合は、カレントディレクトリ下に指定したファイル名で ZIP ファイルが作成されます。

フルパスで指定した場合は、指定したディレクトリ下に指定した名前での ZIP ファイルが作成されます。

拡張子に「.zip」以外を指定した場合も、ZIP 形式で出力されます。

<snapshot ログ収集対象定義ファイル名 >

snapshot ログ収集対象定義ファイルをフルパスで指定します。省略した場合は、次のファイルに定義されたファイルが収集対象になります。空白を含む場合は、"" (ダブルクォーテーション) で囲みます。

- Windows の場合
`<Cosminexus のインストールディレクトリ>%manager%config%snapshotlog.conf`
- UNIX の場合

11. ログの運用で使用するコマンド

```
/opt/Cosminexus/manager/config/snapshotlog.conf
```

なお、snapshot ログ収集対象定義ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」を参照してください。

入力例

カレントディレクトリに出力する場合

```
snapshotlog snapshot.zip
```

指定したディレクトリに出力する場合

- Windows の場合

```
snapshotlog c:¥tmp¥snapshot.zip
```

- UNIX の場合

```
snapshotlog /tmp/snapshot.zip
```

snapshot ログ収集対象定義ファイル名を指定した場合

- Windows の場合

```
snapshotlog snapshot.zip c:¥temp¥snapshotlog.conf
```

- UNIX の場合

```
snapshotlog snapshot.zip /temp/snapshotlog.conf
```

12 監査ログの設定で使用する コマンド

この章では、監査ログの設定で使用するコマンドの入力形式、機能などについて説明します。

12.1 監査ログの設定で使用するコマンドの一覧

12.2 監査ログの設定で使用するコマンドの詳細

12.1 監査ログの設定で使用するコマンドの一覧

監査ログの設定で使用するコマンドの一覧を次の表に示します。

表 12-1 監査ログの設定で使用するコマンドの一覧

コマンド名称	分類	概要
auditsetup	監査ログの設定	監査ログ定義ファイルに指定されたプロパティに従って空の監査ログファイルを出力し、所有者およびアクセス権限を設定します。

12.2 監査ログの設定で使用するコマンドの詳細

監査ログの設定で使用するコマンドの入力形式、機能などを次に示します。

auditsetup (監査ログの設定)

形式

```
auditsetup
```

機能

監査ログ定義ファイルに指定されたプロパティに従って空の監査ログファイルを出力し、所有者およびアクセス権限を設定します。

監査ログ定義ファイルは環境変数 `COSMINEXUS_AUDITLOG_CONF` で指定されているファイルを使用します。環境変数 `COSMINEXUS_AUDITLOG_CONF` が設定されていない場合は、デフォルトの設定ファイルを使用します。デフォルトの設定ファイルは次のとおりです。

Windows の場合

```
<Cosminexus のインストールディレクトリ >%common%\conf\auditlog.properties
```

UNIX の場合

```
/opt/Cosminexus/common/conf/auditlog.properties
```

監査ログ定義ファイルの指定値と設定内容は、Windows の場合と UNIX の場合で異なります。監査ログ定義ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」を参照してください。

引数

なし

戻り値

0 :

正常終了しました。

0 以外 :

異常終了しました。

注意事項

このコマンドを実行したあと、監査ログファイルを削除 (移動も含む) しないでください。削除 (または移動) した場合、監査ログを出力したプロセスのユーザおよびブ

12. 監査ログの設定で使用するコマンド

ライマリグループが監査ログの所有者となります。

このコマンドを再度実行する場合、すべての監査ログファイル、管理ファイル、メッセージログファイルおよび例外ログファイルを削除してから、またはこれらのファイルの出力ディレクトリを変更してから実行してください。これらのファイルが残っている場合、実行に失敗します。

メッセージログファイルの出力ディレクトリが存在しない場合、デフォルトの出力ディレクトリが使用されます。

監査ログ定義ファイルを読み込めない場合や監査ログ出力ディレクトリが存在しない場合、実行に失敗します。

13 統合ユーザ管理で使用する コマンド

この章では、統合ユーザ管理で使用するコマンドの入力形式、機能などについて説明します。

13.1 統合ユーザ管理で使用するコマンドの一覧

13.2 統合ユーザ管理で使用するコマンドの詳細

13.1 統合ユーザ管理で使用するコマンドの一覧

統合ユーザ管理で使用するコマンドの一覧を、次の表に示します。

表 13-1 統合ユーザ管理で使用するコマンドの一覧

コマンド名称	分類	概要
convpw	パスワードの暗号化	ldif ファイル内のパスワードフィールドの内容を暗号化して、その結果を標準出力に出力します。
ssoexport	シングルサインオン情報リポジトリの参照	シングルサインオン情報リポジトリに内容を読み込んで、標準出力に CSV 形式で表示します。
ssogenkey	暗号鍵ファイルの作成	シングルサインオン情報リポジトリに登録 / 参照するときの暗号鍵ファイルを作成します。
ssoimport	シングルサインオン情報リポジトリの登録	シングルサインオン用の認証情報が格納されている CSV 形式のファイルを読み込んでシングルサインオン情報リポジトリに登録します。
uachpw	パスワードの変更	統合ユーザ管理のコンフィグレーションファイル中の LDAP ディレクトリサーバと DB サーバにアクセスするためのパスワードを変更します。

13.2 統合ユーザ管理で使用するコマンドの詳細

統合ユーザ管理で使用する各コマンドの入力形式、機能などを次に示します。

コマンドの格納先

統合ユーザ管理で使用するコマンドは、次のディレクトリに格納されています。

- Windows の場合
`<Cosminexus のインストールディレクトリ>\¥manager¥bin¥`
- UNIX の場合
`/opt/Cosminexus/manager/bin/`

共通の仕様

終了コード

統合ユーザ管理で使用するコマンドの終了コードを、次の表に示します。

表 13-2 統合ユーザ管理で使用するコマンドの終了コード

終了コード	意味
0	正常終了しました。
1	コマンドの実行中にエラーが発生しました。
2	コマンドまたはサーバ起動時の引数に誤りがあります。

convpw (パスワードの暗号化)

形式

```
convpw [-f {md5|sha1}] <ldif_file_name> <password_attribute>
```

機能

ユーザ情報リポジトリに登録するときに、ldif ファイルの内容を暗号化します。このコマンドは、指定した ldif ファイルの内容を読み取って、<password_attribute> で指定されている内容を暗号化して標準出力に出力します。ldif ファイルの内容で、<password_attribute> で指定した属性名の値以外は、そのまま標準出力に出力します。

なお、<password_attribute> で指定した属性名がない場合、そのままファイルの内容が標準出力に出力されます。

なお、このコマンドは、root 権限があるユーザ、またはコマンドの実行権限を設定した特定のユーザだけが実行できます。特定のユーザにコマンドの実行権限を設定する方法については、「7.2 Management Server で使用するコマンドの詳細」の「mngenvsetup (管理グループの設定)」を参照してください。

引数

`-f {md5|sha1}`

<ldif_file_name> に指定したファイルの内容の中で、password (パスワード) で指定された値を暗号化する形式を指定します。省略した場合は、sha1 を仮定します。ここで指定する値は、大文字と小文字を区別しません。

- md5
MD5 形式で暗号化します。
- sha1
SHA-1 形式で暗号化します。

<ldif_file_name>

パスワード変換をするユーザ情報が格納された ldif ファイルのファイル名を指定します。

<password_attribute>

パスワードのフィールドの内容を変換する際の属性名を指定します。

注意事項

変換する ldif ファイルに日本語を記述する場合、UTF-8 に変換したあとで、base64 エンコードしてください。LDIF の詳細については、RFC 2849 "The LDAP Data Interchange Format (LDIF) - Technical Specification" を参照してください。

ssoexport (シングルサインオン情報リポジトリの参照)

形式

```
ssoexport [-n <レルム名>] [-u <ユーザID>] [-scramble]  
<useradmin_configfile>
```

機能

シングルサインオン情報リポジトリに格納されたユーザ情報を、CSV 形式で標準出力に出力します。

シングルサインオン情報リポジトリのユーザ情報を変更する場合、ssoexport コマンドによって出力された情報を CSV 形式ファイルにリダイレクトして、CSV 形式ファイルを編集したあとで、ssoimport コマンドで再度登録処理をします。

ssoexport コマンドでユーザ情報を取り出した場合、項目 " SECRETDATA " の実データは復号化されます。

ssoexport コマンドがユーザ情報を CSV 形式に変換して出力するには、ユーザ情報にレルムとユーザ ID が含まれている必要があります。含まれていない場合、出力しないで無視します。

なお、このコマンドは、root 権限があるユーザ、またはコマンドの実行権限を設定した特定のユーザだけが実行できます。特定のユーザにコマンドの実行権限を設定する方法については、「7.2 Management Server で使用するコマンドの詳細」の「mngenvsetup (管理グループの設定)」を参照してください。

引数

-n <レルム名>

検索するレルム名を指定します。省略した場合、すべてのレルム名が対象となります。

-u <ユーザ ID>

検索するユーザ ID を指定します。ユーザ ID には、ワイルドカード (*) を指定できます。ワイルドカード (*) を指定する場合は、ダブルクォーテーション (") で囲んでください。

(例)

- -u "*"

すべてのユーザを取り出します。
- -u "Ta*"

ユーザ ID が Ta で始まるユーザを取り出します。
- -u "*no"

ユーザ ID が no で終わるユーザを取り出します。

省略した場合、すべてのユーザ ID が対象となります。

-scramble

パスワード変更コマンド (uachpw) でパスワードをスクランブル化した場合に指定します。

<useradmin_configfile>

統合ユーザ管理のコンフィグレーションファイル (ua.conf) を指定します。

入力例・出力例

レルム名が「RealmA」で、「s9」で始まるユーザを取り出す場合の入力例および出力例を次に示します。

入力例

Windows の場合

```
C:¥>ssoexport -n RealmA -u "s9*" "C:¥Program
Files¥Hitachi¥Cosminexus¥manager¥config¥ua.conf"
```

UNIX の場合

13. 統合ユーザ管理で使用するコマンド

```
% ssoexport -n RealmA -u "s9*" /opt/Cosminexus/manager/  
config/ua.conf
```

出力例

```
SecurityDomain, USERID, SECRETDATA, PUBLICDATA, LINK_J2EE, LINK_REAL  
MA  
RealmA, s981234, abfdef, ,  
RealmA, s991234, ghijkl, ,
```

注意事項

シングルサインオン情報リポジトリに情報がない場合、または指定したレルム名 / ユーザ ID に対応したユーザ情報が取得できない場合、ヘッダだけ出力してこのコマンドは終了します。

ssoexport コマンドを実行中に LDAP ディレクトリサーバを停止しないでください。エラーメッセージを出力しないで、コマンドが終了することがあります。

ssoexport コマンドと ssoimport コマンドは、同時に実行しないでください。

ssoimport コマンドで登録する接続先のユーザ管理を持つアプリケーションについては、整合性を取りません。つまり、シングルサインオン情報リポジトリ内に対応するレルムのエントリ（またはユーザエントリ）がなくてもエラーとなりません。この場合、ssoexport コマンドで情報を参照しても出力されないため注意が必要です。

ssoexport コマンドでは、接続先のユーザ管理を持つアプリケーションについては、ベース DN の直下にあるレルムのエントリに対応する値しか表示しません。

ssogenkey (暗号鍵ファイルの作成)

形式

```
ssogenkey <useradmin_configfile>
```

機能

シングルサインオン用の認証情報は、暗号化して保存します。また、参照する際は、復号化して参照します。このコマンドは、暗号化および復号化で使用する暗号鍵を作成します。

なお、このコマンドは、root 権限があるユーザ、またはコマンドの実行権限を設定した特定のユーザだけが実行できます。特定のユーザにコマンドの実行権限を設定する方法については、「7.2 Management Server で使用するコマンドの詳細」の「mngenvsetup (管理グループの設定)」を参照してください。

引数

```
<useradmin_configfile>
```

統合ユーザ管理のコンフィグレーションファイル (ua.conf) を指定します。

注意事項

指定したファイルがすでにある場合は、".n" を付けて、指定したファイルと同一ディレクトリ内にバックアップを取ります。

このコマンドでは、シングルサインオン情報リポジトリの内容はアクセスしません。すでにシングルサインオン情報リポジトリ内にシングルサインオン用の認証情報を登録している場合は、先にすべての情報を `ssoexport` で取り出したあと、`ssoimport` コマンドで登録する必要があります。

ssoimport (シングルサインオン情報リポジトリの登録)

形式

```
ssoimport {-a|-m|-d|-x} [-p] [-scramble] <csvfile_name>
<useradmin_configfile>
```

機能

ユーザ管理を持つアプリケーションから取り出した CSV 形式ファイル (または取り出して編集した CSV 形式ファイル) を、シングルサインオン情報リポジトリに登録します。登録時、項目 ID "SECRETDATA" に記述された実データは暗号化されてシングルサインオン情報リポジトリに登録されます。

なお、このコマンドは、root 権限があるユーザ、またはコマンドの実行権限を設定した特定のユーザだけが実行できます。特定のユーザにコマンドの実行権限を設定する方法については、「7.2 Management Server で使用するコマンドの詳細」の「mngenvsetup (管理グループの設定)」を参照してください。

引数

-a

<csvfile_name> に指定したファイルの内容をシングルサインオン情報リポジトリに追加します。シングルサインオン情報リポジトリ内に追加しようとするユーザのエントリがある場合は、該当ユーザに対する処理はしないで、警告メッセージを出力して次行以降の行の処理を続けます。

-m

<csvfile_name> に指定したファイルの内容でシングルサインオン情報リポジトリを上書きします。シングルサインオン情報リポジトリ内に上書きしようとするユーザのエントリがない場合は、ユーザエントリを追加します。

-d

<csvfile_name> に指定したファイルの内容をシングルサインオン情報リポジトリから削除します。シングルサインオン情報リポジトリ内に削除しようとするユーザのエントリがない場合は、警告メッセージを出力して処理は続けます。

13. 統合ユーザ管理で使用するコマンド

-x

ラインオペレーションの指定に従って、シングルサインオン情報リポジトリの内容を更新します。シングルサインオン用認証情報の CSV 形式ファイルのラインオペレーションについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」を参照してください。

-p

追加、変更、または更新したレール名とユーザ名の一覧を標準出力に出力します。

-scramble

パスワード変更コマンド（uachpw）でパスワードをスクランブル化した場合に指定します。

<csvfile_name>

シングルサインオン情報リポジトリに登録する CSV 形式ファイルを指定します。

<useradmin_configfile>

統合ユーザ管理のコンフィグレーションファイル（ua.conf）を指定します。

入力例

ここでは、登録する CSV 形式ファイルを userdata.csv、シングルサインオン用設定ファイルを "ua.conf" として説明します。

CSV 形式ファイルの内容をシングルサインオン情報リポジトリに追加する場合

```
ssoimport -a userdata.csv ua.conf
```

CSV 形式ファイルの内容をシングルサインオン情報リポジトリから削除する場合

```
ssoimport -d userdata.csv ua.conf
```

CSV 形式ファイル OPERATION 項目の実データに指定したオペレーションに従ってシングルサインオン情報リポジトリに登録する場合

```
ssoimport -x userdata.csv ua.conf
```

出力メッセージ

ssoimport コマンドは、オプションに指定した CSV 形式ファイルから 1 行ずつ取得してシングルサインオン情報リポジトリに登録（変更または削除）します。このとき、-p オプションを指定して実行すると、その操作についての情報を標準出力に出力します。なお、操作時に発生した警告、エラーメッセージについては、標準エラー出力に出力します。

-p オプションを指定しないで実行した場合は、警告、エラーメッセージと「結果」だけを出力します。

出力例を次の図に示します。

図 13-1 ssoimport コマンドの出力例 (Windows の場合)

```

c:\> ssoimport -x -p xxx.csv .\ssoconf.conf
OPERATION  REALMNAME  USERID
-----
add        Portal    USER1
delete     Portal    USER3
xxxx0010-W Not Found User-ID, line=xxx realm=Portal uid=WWW
:
: (中略)
:
modify     Portal    jirou
-----
Total:100 ADD:90 MODIFY:8 DELETE:1 WARNING:1
c:\>

```

図 13-1 の出力例は、Windows のコマンドプロンプトで実行された ssoimport コマンドの結果を示しています。出力は表形式で表示され、タイトル行 (OPERATION, REALMNAME, USERID) が破線の下にあり、実行内容 (add, delete, modify) が破線の間にあり、結果行 (Total:100 ADD:90 MODIFY:8 DELETE:1 WARNING:1) が破線の後に表示されます。右側の注釈は、タイトル行を「タイトル」、実行内容を「実行内容」、結果行を「結果」として括弧で括弧しています。

図 13-2 ssoimport コマンドの出力例 (UNIX の場合)

```

% ssoimport -x -p xxx.csv ./ssoconf.conf
OPERATION  REALMNAME  USERID
-----
add        Portal    USER1
delete     Portal    USER3
xxxx0010-W Not Found User-ID, line=xxx realm=Portal uid=WWW
:
: (中略)
:
modify     Portal    jirou
-----
Total:100 ADD:90 MODIFY:8 DELETE:1 WARNING:1
%

```

図 13-2 の出力例は、UNIX のコマンドプロンプトで実行された ssoimport コマンドの結果を示しています。出力は表形式で表示され、タイトル行 (OPERATION, REALMNAME, USERID) が破線の下にあり、実行内容 (add, delete, modify) が破線の間にあり、結果行 (Total:100 ADD:90 MODIFY:8 DELETE:1 WARNING:1) が破線の後に表示されます。右側の注釈は、タイトル行を「タイトル」、実行内容を「実行内容」、結果行を「結果」として括弧で括弧しています。

「OPERATION」「REALMNAME」「USERID」の順に表示します。

実行内容

ヘッダに対応した「OPERATION」「REALMNAME」「USERID」の内容、および操作時に発生した警告/エラーメッセージを表示します。

OPERATION

次の表に示す操作種別のどれかを出力します。

表 13-3 ssoimport コマンドの操作種別

項目	内容
add	情報を追加したことを意味します。
modify	情報を変更 (上書き) したことを意味します。
delete	情報を削除したことを意味します。

13. 統合ユーザ管理で使用するコマンド

REALMNAME

操作の対象となるレルム名を出力します。この値は、CSV 形式ファイルの「REALMNAME」項目に指定した値を表示します。

USERID

操作の対象となるユーザ ID を出力します。この値は、CSV 形式ファイルのヘッダ「USERID」項目に指定した値を表示します。

実行結果

次の表に示す実行結果を表示します。

表 13-4 ssoimport コマンドの実行結果

項目	内容
Total	操作の対象となった行数を出力します。
ADD	シングルサインオン情報リポジトリ内に追加したエントリ数を表示します。
MODIFY	シングルサインオン情報リポジトリ内を変更したエントリ数を表示します。
DELETE	シングルサインオン情報リポジトリから削除したエントリ数を表示します。
WARNING	上記の操作をした際に発生した警告メッセージの数を表示します。

注意事項

接続先の JAAS 対応ユーザ管理を持つアプリケーションのユーザの情報を削除する場合は、ssoexport で対象とするユーザを取得したあとで、削除したい接続先のユーザ管理を持つアプリケーションのユーザを削除し、-m オプションで更新してください。

ssoexport コマンドと ssoimport コマンドは、同時に実行しないでください。

ssoimport コマンドで登録する接続先のユーザ管理を持つアプリケーションについては、整合性を取りません。つまり、シングルサインオン情報リポジトリ内に対応するレルムのエントリ（またはユーザエントリ）がなくてもエラーとなりません。この場合、ssoexport コマンドで情報を参照しても出力されないため注意が必要です。

ssoexport コマンドでは、接続先のユーザ管理を持つアプリケーションについては、ベース DN の直下にあるレルムのエントリに対応する値しか表示しません。

uachpw (パスワードの変更)

形式

```
uachpw [-scramble] [-ldap.<n> <パスワード>] [-db.<n> <パスワード>]  
<useradmin_configfile>
```

機能

統合ユーザ管理のコンフィグレーションファイル (ua.conf) の LDAP ディレクトリサーバと DB サーバにアクセスするためのパスワードを変更します。また、変更時にパス

ワードをスクランブル化できます。

なお、このコマンドは、root 権限があるユーザ、またはコマンドの実行権限を設定した特定のユーザだけが実行できます。特定のユーザにコマンドの実行権限の設定については、「7.2 Management Server で使用するコマンドの詳細」の「mngenvsetup (管理グループの設定)」を参照してください。

引数

-scramble

変更するパスワードをスクランブル化する場合に指定します。

-ldap.<n> <パスワード>

<n> で指定した LDAP ディレクトリサーバにアクセスするためのパスワードを変更します。<n> には、統合ユーザ管理のコンフィグレーションファイルに定義した LDAP 設定番号を指定します。<パスワード> には、変更する新しいパスワードを指定します。

-db.<n> <パスワード>

<n> で指定した DB サーバにアクセスするためのパスワードを変更します。<n> には、統合ユーザ管理のコンフィグレーションファイルに定義した JDBC 設定番号を指定します。<パスワード> には、変更する新しいパスワードを指定します。

<useradmin_configfile>

統合ユーザ管理のコンフィグレーションファイル (ua.conf) を指定します。この引数は省略できません。

入力例

統合ユーザ管理のコンフィグレーションファイル (ua.conf) に定義したパスワードを下記のように変更し、-scramble オプションを使用してスクランブル化する場合の入力例を示します。

- LDAP アクセス情報 0 のパスワードを diradmin に変更
- LDAP アクセス情報 1 のパスワードを administrator に変更
- DB アクセス情報 0 を tiger に変更

```
% uachpw -scramble -ldap.0 diradmin -ldap.1 administrator -db.0
tiger ua.conf
```

注意事項

-scramble オプションでパスワードをスクランブル化する場合は、usrconf.properties の com.cosminexus.admin.auth.passwordScramble.enable に true を設定してください。設定しなかった場合、パスワードが復号化されないため LDAP ディレクトリサーバや DB にアクセスできません。

13. 統合ユーザ管理で使用するコマンド

パスワードに使用できる文字数は 30 文字までです。

オプションの大文字と小文字は区別されません。

コマンドを二重で実行しないでください。

引数に統合ユーザ管理のコンフィグレーションファイル (ua.conf) 以外のファイルは指定しないでください。

14 Java アプリケーションで使用するコマンド

この章では、Java アプリケーションで使用するコマンドの入力形式、機能などについて説明します。

14.1 Java アプリケーションで使用するコマンドの一覧

14.2 Java アプリケーションで使用するコマンドの詳細

14.1 Java アプリケーションで使用するコマンドの一覧

Java アプリケーションで使用するコマンドの一覧を次の表に示します。

表 14-1 Java アプリケーションで使用するコマンドの一覧

コマンド名称	分類	概要
cjldellog	Java アプリケーションのシステムログ出力先サブディレクトリの削除	vbj コマンドで起動した Java アプリケーションのシステムログ出力先サブディレクトリを削除します。
cjldumpap	Java アプリケーションのスレッドダンプの取得	cjclstartap コマンドで開始した Java アプリケーションで障害が発生した場合に、JavaVM のスレッドダンプを取得します。
cjclstartap	Java アプリケーションの開始	Java アプリケーションを開始します。

14.2 Java アプリケーションで使用するコマンドの詳細

Java アプリケーションで使用する各コマンドの入力形式、機能などを次に示します。

コマンドの格納先

Java アプリケーションで使用するコマンドは、次のディレクトリに格納されています。

- Windows の場合

<Cosminexus のインストールディレクトリ>%CC%\client\bin%

- UNIX の場合

/opt/Cosminexus/CC/client/bin/

注

uCosminexus Client を使用している場合は、格納ディレクトリの

「<Cosminexus のインストールディレクトリ>%CC%」を、「<Cosminexus のインストールディレクトリ>%CCL%」と読み替えてください。

cjcldellog (Java アプリケーションのシステムログ出力先サブディレクトリの削除)

形式

```
cjcldellog -t <時間>h|<日数>d [-f] <ログ出力先ディレクトリ名称>
[<ログ出力先ディレクトリ名称> ...]
```

機能

引数に指定したログ出力先ディレクトリに、指定した日数または時間より前の更新日時のサブディレクトリがあった場合に、削除します。なお、`-f` を指定しない場合は、起動中の Java アプリケーションのサブディレクトリは削除しません。

なお、このコマンドは、`vbj` コマンドで起動した Java 環境で実行するコマンドです。

このコマンドは、タスク (Windows の場合)、または `at` コマンドや `crontab` コマンド (UNIX の場合) を使用して定期的に行ってください。

引数

`-t <時間>h`

更新時間が何時間よりも前のサブディレクトリを削除するかを 1 ~ 8760 の範囲 (単位: 時間) で指定します。例えば、2 を指定した場合、更新時間が 2 時間前よりも古いサブディレクトリが削除されます。

14. Java アプリケーションで使用するコマンド

-t <日数> d

更新日が何日より前のサブディレクトリを削除するかを 1 ~ 365 の範囲 (単位: 日) で指定します。例えば, 2 を指定した場合, 更新日が 2 日より古いサブディレクトリが削除されます。

-f

指定された日時より古いサブディレクトリすべてを削除対象とします。省略した場合は, client_process_end ファイルが存在するサブディレクトリだけが削除対象になります。

この引数を使用すると, Java アプリケーション起動中のサブディレクトリも削除対象としてしまうおそれがあるため, すべての Java アプリケーションが終了していることを確認してからコマンドを実行してください。

<ログ出力先ディレクトリ名称>

削除対象にするサブディレクトリの上位ディレクトリである, ログ出力先ディレクトリを, フルパス名で指定します。ログ出力先ディレクトリを複数指定するときは, 空白で区切ってください。また, 複数のログ出力先ディレクトリを指定した場合, 最後のディレクトリだけにワイルドカード (*) を指定できます。

なお, Windows の場合, 空白を含むパスを指定するときは, パス全体を "" で囲む必要があります。

指定できるパスの例を次に示します。

- Windows の場合

```
"C:¥Program Files¥Hitachi¥Cosminexus¥CC¥client¥logs¥ejb"  
C:¥work¥logs¥ejb*  
C:¥work¥logs¥ejb*log  
C:¥work¥logs¥*log
```

- UNIX の場合

```
/work/logs/ejb*  
/work/logs/ejb*log  
/work/logs/*log
```

戻り値

0:

正常終了しました。

1:

異常終了しました。

入力例

7 日より前の更新日時サブディレクトリをすべて削除する場合

- Windows の場合

```
cjclldellog -t 7d -f c:¥work¥logs
```

- UNIX の場合

```
cjcldellog -t 7d -f /work/logs
```

12 時間より前の更新日時のサブディレクトリで client_process_end ファイルが存在するサブディレクトリを削除する場合

- Windows の場合

```
cjcldellog -t 12h c:¥work¥logs
```

- UNIX の場合

```
cjcldellog -t 12h /work/logs
```

複数のディレクトリ下にある 12 時間より前の更新日時のサブディレクトリで、client_process_end ファイルが存在するサブディレクトリを削除する場合

- Windows の場合

```
cjcldellog -t 12h c:¥work¥logs¥ejb1 c:¥work¥logs¥ejb2
c:¥work¥logs¥ejb3
```

- UNIX の場合

```
cjcldellog -t 12h /work/logs/ejb1 /work/logs/ejb2 /work/logs/
ejb3
```

“ejb” で始まるディレクトリにある 12 時間より前の更新日時のサブディレクトリで、client_process_end ファイルが存在するサブディレクトリを削除する場合

- Windows の場合

```
cjcldellog -t 12h c:¥work¥logs¥ejb*
```

- UNIX の場合

```
cjcldellog -t 12h /work/logs/ejb*
```

注意事項

ログ出力サブディレクトリ配下にユーザがディレクトリを作成した場合、このコマンドではサブディレクトリ下のログファイルは削除しますが、ユーザ作成ディレクトリは削除しません。この場合、サブディレクトリとユーザ作成ディレクトリが削除されないで残ります。

このコマンドでは作成後 1 時間以上経過していないログは削除できません。

サブディレクトリ共有モードで動作しているアプリケーションが使用しているサブディレクトリを削除しないでください。削除した場合、以降のログ出力が正しく行われなくなることがあります。

cjcdumpap (Java アプリケーションのスレッドダンプの取得)

形式

`cjcdumpap` [<プロセスID>]

機能

`cjclstartap` コマンドで開始した Java アプリケーションのスレッドダンプを、`cjclstartap` コマンドを実行しているカレントディレクトリに出力します。また、指定した `cjclstartap` プロセスのスレッドダンプを出力することもできます。

プロセスを指定しない場合、スレッドダンプを出力したい `cjclstartap` コマンドを実行しているカレントディレクトリで、このコマンドを実行してください。

引数

<プロセス ID>

特定のプロセスのスレッドダンプを出力する場合、`cjclstartap` コマンドのプロセス ID を指定します。引数を指定しない場合、`cjcdumpap` コマンドを実行したカレントディレクトリから起動した `cjclstartap` プロセスに対するスレッドダンプを出力します。存在しないプロセス ID を指定した場合、エラーメッセージが出力されて、異常終了します。

プロセス ID 以外を指定した場合、コマンドの使用方法を表示します。

戻り値

0:

正常終了しました。

1:

異常終了しました。

注意事項

`cjcdumpap` コマンドは、`cjclstartap` コマンドを実行したユーザ、またはスーパーユーザが実行できます。

`usrconf.cfg` の `ejb.client.directory.shareable` に「false」を指定していないで `cjclstartap` コマンドを実行した場合、引数に <プロセス ID> を指定しないで `cjcdumpap` コマンドを実行すると、エラーメッセージが出力されてスレッドダンプが出力されないことがあります。

cjclstartap (Java アプリケーションの開始)

形式

```
cjclstartap [[<Javaオプション>]...] <Javaアプリケーションクラス名>
            [[<mainメソッドに渡される引数>] ...]
```

機能

Java アプリケーションを開始します。

引数

引数の指定がない場合、コマンドの使用方法を表示します。

<Java オプション >

JavaVM の起動オプションを指定します。

<Java オプション > とオプション定義ファイル (usrconf.cfg) で同じオプションを指定した場合、<Java オプション > に指定した内容が有効になります。

指定できるオプションを次に示します。

- -cp
- -classpath
- -D<name>=<value>
- -verbose[:class | gc | jni]
- -ea[:<packagename>... | :<classname>]
- -enableassertions[:<packagename>... | :<classname>]
- -da[:<packagename>... | :<classname>]
- -disableassertions[:<packagename>... | :<classname>]
- -esa | -enablesystemassertions
- -dsa | -disablesystemassertions
- -agentlib:<libname>[=<options>]
- -agentpath:<pathname>[=<options>]
- -javaagent:<jarpath>[=<options>]
- -Xmixed
- -Xint
- -Xbootclasspath:<directories and zip/jar files separated by ;>
- -Xbootclasspath/a:<directories and zip/jar files separated by ;>
- -Xbootclasspath/p:<directories and zip/jar files separated by ;>
- -Xnoclassgc
- -Xloggc:<file>
- -Xbatch
- -Xms<size>
- -Xmx<size>

14. Java アプリケーションで使用するコマンド

- -Xmn<size>
- -Xss<size>
- -Xprof
- -Xrunhprof[:help] |[:<option>=<value>, ...]
- -Xdebug
- -Xrs
- -Xcheck:jni
- -XX:NewRatio=<value>
- -XX:PermSize=<size>
- -XX:MaxPermSize=<size>
- -XX:SurvivorRatio=<value>
- -XX:[+|-]PrintTenuringDistribution
- -XX:TargetSurvivorRatio=<value>
- -XX:MaxTenuringThreshold=<value>

なお、次のオプションは指定できません。

- -jar jarfile (-cp jarfile <Java アプリケーションクラス名> で代替)
- -client (usrconf.cfg ファイルに jvm.type=client を指定することで代替)
- -server (usrconf.cfg ファイルに jvm.type=server を指定することで代替)
- -hotspot
- -version
- -showversion
- -? -help
- -X
- -XX
- -Xfuture

<Java アプリケーションクラス名>

ユーザ作成の Java アプリケーションクラス名を指定します。

<main メソッドに渡される引数>

Java アプリケーションの main メソッドに渡す引数を指定します。

戻り値

次の戻り値以外は、Java アプリケーションの終了コードが返却されます。ただし、UNIX では SIGHUP, SIGINT, SIGTERM が発生した場合、シグナル番号に +128 した値が返却されます。

1 :

Java アプリケーション開始コマンドでエラーが発生しました。

143 :

コマンド実行中に次の操作をしました (Windows の場合)。

- コマンドを起動しているコマンドプロンプトのウィンドウを「×」ボタンを押下して閉じた。
- Windows システムをシャットダウンした。
- Windows システムをログオフした。

注意事項

オプション定義ファイル (usrconf.cfg)、およびユーザプロパティファイル (usrconf.properties) が格納されているディレクトリパスを指定する場合、ユーザ定義ファイル格納先環境変数 (CJCLUSRCONFDIR) を指定する必要があります。Java アプリケーションの環境変数の設定については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」を参照してください。

ユーザ定義ファイル格納先環境変数が設定されていない場合、カレントディレクトリが参照されます。環境変数で設定したディレクトリにオプション定義ファイル (usrconf.cfg)、またはユーザプロパティファイル (usrconf.properties) が存在しない場合は、エラーメッセージが出力され、異常終了します。

このコマンドは、Java アプリケーションのメインクラスの main メソッドを実行します。main メソッドは、public static void main(String[]) と宣言する必要があります。

CLASSPATH 環境変数に指定したクラスパスは無視されます。

このコマンドを実行するには、カレントディレクトリ、およびログ出力先ディレクトリに、書き込み権限が必要です。

UNIX の場合、ログの出力先を複数のユーザで使用するには、umask を「0」に設定しておく必要があります。

ejb.client.directory.shareable オプションに「false」を指定した場合、同じカレントディレクトリ、または同じログ出力先ディレクトリを使用する複数のプロセスを同時に起動しようとすると、エラーメッセージが出力されて、異常終了します。

ejb.client.directory.shareable オプションに「false」を指定した場合、カレントディレクトリにプロセス ID を格納するファイル (.cjlstartap.pid) とロックファイル (.cjlstartap.lock) が作成されます。また、UNIX の場合、パイプファイル (.COSMINEXUS_CC_EJBCLIENT_<プロセス ID>) が作成されます。これらのファイルを編集または削除した場合、ログファイル、およびスレッドダンプが正しく出力されないことがあります。

JavaVM イベントログの「hs_err<pid>.log」、および core ダンプはカレントディレクトリに出力されます。

Windows の場合、UNC 名を含むパスをカレントディレクトリにすると、ログが出力されないことがあります。

UNIX の場合、nfs マウントされたディスクをカレントディレクトリにすると、ログが出力されないことがあります。

14. Java アプリケーションで使用するコマンド

usrconf.cfg の `ejb.client.directory.shareable` オプションに「false」を指定しないで、同じログ出力先を指定した複数の `ejclstartap` コマンドから同時に Java ログが出力された場合、Java ログが正しく出力されないことがあります。その場合、`ejb.client.log.directory` に指定するディレクトリパスは別にしてください。

`-cp` オプション、および `-classpath` オプションの指定がない場合、デフォルトのクラスパスとしてカレントディレクトリを使用しません。

`-cp` オプション、および `-classpath` オプションを複数指定した場合、すべての指定が有効になります。

Java アプリケーションログはサブディレクトリ共有モードだけで使用できます。サブディレクトリ占有モードに設定した場合、ログは正しく出力されません。

アカウントのログオフ時にこのコマンドを実行した場合、起動に失敗することがあります。

このコマンドは、スレッドダンプを出力するため、`Ctrl+Break` の信号を自身に送信します。この場合、`ejclstartap` コマンドとコンソールを共有する `Ctrl+Break` を無視していないアプリケーションは終了されます。

Windows Server 2008 x86、Windows Server 2008 x64、Windows Server 2008 R2、Windows 7 または Windows Vista の場合、このコマンドはユーザに書き込み権限があるディレクトリで実行してください。一般ユーザ権限で、`%ProgramFiles%` およびそのサブディレクトリで実行した場合、カレントディレクトリがリダイレクトされるため次のように動作します。

- ログファイル、ロックファイル、pid ファイルがリダイレクトされます。
- 複数のユーザが実行した場合、ユーザごとにプロセスが同時に動作します。
- このコマンドを実行したユーザと異なるユーザは、`ejldumpap` コマンドを実行できません。

15 Web アプリケーションで使用するコマンド

この章では、Web アプリケーションで使用するコマンドの入力形式、機能などについて説明します。

15.1 Web アプリケーションで使用するコマンドの一覧

15.2 Web アプリケーションで使用するコマンドの詳細

15.1 Web アプリケーションで使用するコマンドの一覧

Web アプリケーションで使用するコマンドの一覧を次の表に示します。

表 15-1 Web アプリケーションで使用するコマンドの一覧

コマンド名称	分類	概要
cjjspc	JSP の事前コンパイル	JSP ファイルから Java ソースファイルを生成およびコンパイルし、クラスファイルを生成します。
cjjsp2java	JSP ソースのチェック	JSP ファイルのシンタックスチェックをします。

15.2 Web アプリケーションで使用するコマンドの詳細

Web アプリケーションで使用するコマンドの入力形式、機能などを次に示します。

コマンドの格納先

Web アプリケーションで使用するコマンドは、次のディレクトリに格納されています。

- Windows の場合
`<Cosminexus のインストールディレクトリ>%CC%\web\bin%`
- UNIX の場合
`/opt/Cosminexus/CC/web/bin/`

cjjspc (JSP の事前コンパイル)

形式

Windows の場合

```
cjjspc [-jspworkdir <JSPワークディレクトリ名>]
        [-classpath <クラスパス1>[;<クラスパス2>...]]
        [-excludefile <JSPファイルパス1>[;<JSPファイルパス2>...]]
        [-excludelist <コンパイル対象外リストファイルパス>]
        [-keepgenerated]
        [-source {1.3|1.4|1.5|5}]
        [-pageencoding <文字エンコーディング>]
        [-resultlist <実行結果リストファイルパス>]
        [-customactiondeclarevariable]
        [-customactionignorecaseattributename]
        [-usebeannocheckclass]
        [-usebeannocheckduplicateid]
        [-tagnocheckrtexprvalue]
        [-tagrtexprvalueterminate]
        [-taglibnocheckprefix]
        [-debugging]
        [-jsppagedisablebom]
        [-nojavaeetaglib]
        [-lowerversion {2.4|2.5}]
        -root <Webアプリケーションのルートディレクトリ>
        [<JSPファイルパス>...]
```

UNIX の場合

```
cjjspc [-jspworkdir <JSPワークディレクトリ名>]
        [-classpath <クラスパス1>[:<クラスパス2>...]]
        [-excludefile <JSPファイルパス1>[:<JSPファイルパス2>...]]
        [-excludelist <コンパイル対象外リストファイルパス>]
        [-keepgenerated]
        [-source {1.3|1.4|1.5|5}]
        [-pageencoding <文字エンコーディング>]
```

15. Web アプリケーションで使用するコマンド

```
[-resultlist <実行結果リストファイルパス>]
[-customactiondeclarevariable]
[-customactionignorecaseattributename]
[-usebeannocheckclass]
[-usebeannocheckduplicateid]
[-tagnocheckrtexprvalue]
[-tagrtexprvalueterminate]
[-taglibnocheckprefix]
[-debugging]
[-jsppagedisablebom]
[-nojaveetaglib]
[-lowerversion {2.4|2.5}]
-root <Webアプリケーションのルートディレクトリ>
<JSPファイルパス>...
```

機能

JSP ファイルから Java ソースファイルを生成およびコンパイルし、クラスファイルを生成します。このコマンドを実行したときのログは、標準出力または標準エラー出力します。Web アプリケーションに含まれるすべての JSP ファイルをコンパイルできます。

-excludelist オプションまたは -excludefile オプションを指定することで、コンパイルする必要のない JSP ファイルをコンパイル対象外ファイルとして指定できます。コンパイル対象外ファイルには、次の「コンパイルが不要なファイル」または「再度コンパイルする必要がないファイル」のどちらかの条件に当てはまるファイルを指定してください。

次のどれかに該当するファイルでコンパイルが不要なファイル

- 拡張子が「.jsp」または「.jspx」のファイル
- Deployment Descriptor (/WEB-INF/web.xml) の <jsp-file> に指定されたファイル
- Deployment Descriptor (/WEB-INF/web.xml) の
 <jsp-property-group><url-pattern> タグに合致するファイル
- cjjspc コマンドの引数に指定したファイル

コンパイル済みであり、再度コンパイルする必要がないファイル

なお、-excludelist オプションと -excludefile オプションの併用はできません。併用した場合はエラーメッセージが表示され、処理が終了します。

また、-resultlist オプションと -excludelist オプションに同じファイルを指定することで、一度コンパイルに成功した JSP ファイルをコンパイル対象外ファイルに指定できます。また、コンパイルに失敗した JSP ファイルがコンパイル不要な場合は、実行結果リストファイルに出力されたコンパイルに失敗した JSP ファイルパスの行頭の「#」を削除することで、コンパイル対象外に指定できます。

引数

-jspworkdir <JSP ワークディレクトリ名 >

JSP ワークディレクトリ (<Web アプリケーションのルートディレクトリ >

WEB-INF/cosminexus_jsp_work) のディレクトリ名を変更する場合に指定します。

JSP ワークディレクトリ名に使用できる文字列は、A ~ Z, a ~ z, 0 ~ 9 の半角英数字と「_ (半角アンダースコア)」だけです。また、次のディレクトリ名は、大文字・小文字の区別なく、JSP ワークディレクトリ名として使用できません。

- lib
- tags
- classes

-classpath < クラスパス >

JSP のコンパイルに必要なとなるクラスのクラスパスを絶対パスで指定します。デフォルトで指定されているクラスパスは、次のとおりです。

- <Web アプリケーションのルートディレクトリ>/WEB-INF/classes
- <Web アプリケーションのルートディレクトリ>/WEB-INF/lib に含まれる jar ファイル
- Servlet 2.5 API などを含んだアプリケーションサーバのシステムが使用するクラスライブラリ

複数のクラスパスを指定する場合、Windows では「;(半角セミコロン)」, UNIX では「:(半角コロン)」で区切って指定します。なお、アプリケーションサーバのシステムが使用するクラスライブラリ内のクラスファイルと -classpath オプションに指定されたクラスパスに含まれる jar ファイル内のクラスファイルが重複している場合は、アプリケーションサーバのシステムが使用するクラスライブラリ内のクラスファイルが優先して使用されます。

-excludefile <JSP ファイルパス >

コンパイル対象外とする JSP ファイルを指定します。

複数の JSP ファイルパスを指定する場合、Windows では「;(半角セミコロン)」, UNIX では「:(半角コロン)」で区切って指定します。JSP ファイルは、Web アプリケーションのルートディレクトリからの相対パスを指定します。ファイルパスにワイルドカード文字「*(アスタリスク)」や空白文字列を含める場合は、ファイルパスを必ず「" (ダブルクォーテーション)」で囲みます。

ファイルパスに区切り文字 (Windows の場合「;(半角セミコロン)」, UNIX の場合「:(半角コロン)」) を含めることはできません。ファイルパスに区切り文字を含める必要がある場合は、コンパイル対象外リストファイルを使用してください。コンパイル対象外リストは -excludelist オプションで指定します。

指定方法を次に示します。

指定方法 (Windows の場合)

```
cjjspc -excludefile <JSPファイルパス1>;<JSPファイルパス2>;...
-root <Webアプリケーションのルートディレクトリ>
```

指定方法 (UNIX の場合)

```
cjjspc -excludefile <JSPファイルパス1>:<JSPファイルパス2>:...
-root <Webアプリケーションのルートディレクトリ>
```

-excludelist < コンパイル対象外リストファイルパス >

15. Web アプリケーションで使用するコマンド

コンパイル対象外とする JSP ファイルを記述したコンパイル対象外リストファイルを指定します。コンパイル対象外リストファイルとは、コマンドによる JSP 事前コンパイルを実行する際、コンパイルの対象外とする JSP ファイルパスを記述するためのテキストファイルです。

ファイルは相対パスまたは絶対パスで指定します。

指定方法とファイル形式を次に示します。

コンパイル対象外リストファイルの指定方法

```
cjjspc -excludelist <コンパイル対象外リストファイルのパス> -root  
<Webアプリケーションのルートディレクトリ>
```

コンパイル対象外リストファイルのファイル形式

```
# コメント  
<コンパイルの対象外とするJSPファイルパス 1 >  
<コンパイルの対象外とするJSPファイルパス 2 >  
...
```

-keepgenerated

JSP をコンパイルする際、JSP トランスレーション時に生成された Java ソースファイルを残す場合に指定します。作成された Java ソースファイルは、コンパイルされたクラスファイルと同じディレクトリに作成されます。

指定方法を次に示します。

指定方法

```
cjjspc -keepgenerated -root <Webアプリケーションのルートディレクトリ>
```

-source {1.3|1.4|1.5|5}

JSP をコンパイルする際、JSP トランスレーション時に Java ソースファイルが生成されます。この Java ソースファイルをコンパイルする Java 言語仕様のバージョンを指定します。例えば、JSP ファイルに、J2SE1.3 の言語仕様に依存したプログラムコードを記述している場合は、このオプションに 1.3 を指定してください。このオプションを省略した場合、インストールしている JDK の Java 言語仕様に従ってコンパイルします。

指定方法を次に示します。

指定方法

```
cjjspc -source <バージョン> -root <Webアプリケーションのルートディレクトリ>
```

-pageencoding <文字エンコーディング>

JSP ファイルのデフォルトの文字エンコーディングを指定します。デフォルトの文字エンコーディングとして設定できる文字は、JavaVM がサポートしている文字エンコーディングとなります。JavaVM がサポートしている文字エンコーディングについては、JDK のドキュメントのサポートされているエンコーディングに関する説明を参照してください。

また、指定できる文字列は、java.nio API 用の正準名と java.lang API 用の正準名に記載されている文字エンコーディング、およびそれらの別名になります。サポート

されていない文字エンコーディングを指定した場合、エラーメッセージが出力されて処理が終了します。

-resultlist < 実行結果リストファイルパス >

コンパイル実行結果リストファイルを絶対パスまたはカレントディレクトリからの相対パスで指定します。コンパイル実行結果リストファイルには次のリストが出力されます。

- コンパイル対象外の JSP ファイルパス
- コンパイルに成功した JSP ファイルパス
- コンパイルに失敗した JSP ファイルパス

指定方法と出力形式を次に示します。

指定方法

```
cjjspc -resultlist <実行結果リストファイルのパス> -root <Webアプリケーションのルートディレクトリ>
```

実行結果リストファイルの出力形式

```
# Excluded JSP files.
<コンパイル対象外JSPファイルの一覧>
# Compiled JSP files.
<コンパイルに成功したJSPファイル>
# <コンパイルに失敗したJSPファイル>
...
```

指定したファイルと同名のファイルが存在した場合は、ファイルを上書きして処理を続行します。

-customactiondeclarevariable

複数のカスタムタグで、スクリプト変数名とスクリプト変数のスコープが重複する場合、JSP ファイルから生成された Java コードで、2 回目のカスタムタグに対応するスクリプト変数の変数宣言を出力するときに、このオプションを指定します。このオプションを指定しない場合は、スクリプト変数の変数宣言を出力しません。

(*) スクリプト変数のスコープは、`javax.servlet.jsp.tagext.TagExtraInfo` クラスのサブクラス、または TLD ファイルの `variable` 要素内の `scope` 要素で指定できます。

-customactionignorecaseattributename

カスタムタグで指定された属性が、TLD ファイルまたはタグファイルで定義されているかどうかのチェックで、大文字小文字を区別しない場合に指定します。

-usebeannocheckclass

JSP コンパイル時に `<jsp:useBean>` タグの `class` 属性値の検証をしない場合に指定します。

-usebeannocheckduplicateid

JSP コンパイル時に `<jsp:useBean>` タグの `id` 属性値の検証をしない場合に指定します。

15. Web アプリケーションで使用するコマンド

-tagnocheckrtexprvalue

Expression が指定できないタグの属性値に Expression が指定されているかどうかを検証しない場合に指定します。

-tagrtexprvalueterminate

タグの属性値が、「<%=」(または「'<%=」)で開始していて、「%>」(または「%>')」で終了していない属性値がある場合、「"」(または「'」)で囲まれた値を文字列として扱うときに指定します。

-taglibnocheckprefix

aglib ディレクティブの前に、taglib ディレクティブで指定した prefix を使用したカスタムタグを記述しているかどうかをチェックしない場合に指定します。

-debugging

JSP デバッグ機能を有効にする場合に指定します。

JSP デバッグ機能が有効な場合、JSP ファイルから作成されたサーブレットの class ファイルに SMAP が埋め込まれます。また、このオプションを指定した場合と指定しない場合で、出力されるクラス名が異なります。詳細については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 基本・開発編 (Web コンテナ)」を参照してください。

-jspagedisablebom

Servlet 2.5 仕様以降の Web アプリケーションの JSP ページで BOM による文字エンコーディングの決定をしない場合に指定します。

-nojavaetaglib

Servlet 2.5 仕様以降の Web アプリケーションで Java EE 仕様に含まれるタグライブラリ (JSTL および JSF) を自動的にマッピングしない場合に指定します。

-lowerversion {2.4|2.5}

設定した Web アプリケーションのバージョンで、JSP ファイルをコンパイルできません。

指定方法を次に示します。

指定方法

```
cjjspc -lowerversion <バージョン> -root <Webアプリケーションの  
ルートディレクトリ>
```

-root <Web アプリケーションのルートディレクトリ>

コンパイル対象の JSP を含む Web アプリケーションのルートディレクトリを指定します。カレントディレクトリからの相対パスまたは絶対パスで指定します。パス名に空白が含まれるときは、「" (ダブルクォーテーション)」で囲みます。

<JSP ファイルパス>...

JSP ファイル単位のコンパイルを実行する場合に指定します。JSP ファイルパスは、

-root で指定した Web アプリケーションのルートディレクトリからの相対パスで指定します。パスの区切り文字には「/」を使用して、パスの先頭には必ず「/」を付けます。JSP ファイルパスを複数指定する場合は、それぞれの JSP ファイルパスを半角スペースで区切って、複数の JSP ファイルパスを指定します。省略した場合は、Web アプリケーション内のすべての JSP ファイルがコンパイル対象となります。パス名にアスタリスク (*), または空白が含まれるときは、JSP ファイルパスごとに「" (ダブルクォーテーション)」で囲みます。

戻り値

- 0 :
正常終了しました。
- 1 :
オプション解析エラーです。
- 2 以上 :
異常終了しました。

注意事項

このコマンドを実行したときには、Deployment Descriptor (/WEB-INF/web.xml) および TLD ファイルが、DTD または XML スキーマに従っているか検証します。Deployment Descriptor (/WEB-INF/web.xml) がスキーマに従っていない場合は、エラーとなりコマンドの処理が終了します。また、TLD ファイルがスキーマに従っていない場合も、JSP がトランスレーションしているときにエラーとなります。したがって、スキーマに従っていない場合は、エラーとなりコマンドの処理が終了します。タグファイル、静的インクルードされたファイル、または TLD ファイルを更新した場合、更新したファイルを参照するすべての JSP ファイルをコンパイルしてください。

cjjsp2java (JSP ソースのチェック)

形式

```
cjjsp2java [-d <ディレクトリ>|-d:non] [-classpath <クラスパス>]
           [-root <ディレクトリ> [[-taglib <uri> <location>]...]]
           <JSPファイル名>...
```

機能

JSP1.1 仕様および JSP1.2 仕様に準拠した JSP ファイルから Java ソースを生成し、JSP ファイルのシンタックスチェックをします。

なお、JSP2.0 仕様以降に準拠した JSP ファイルのシンタックスチェックをする場合は、

ejjspc コマンドを使用してください。

引数

-d <ディレクトリ>

生成した java ソースファイルの出力先ディレクトリを指定します。このオプションを指定しなかった場合は、コマンドを実行したカレントディレクトリ下に生成されます。

-root オプションを指定していない場合、-d オプションで指定したディレクトリの直下に java ソースファイルが生成されます。同一ファイル名の JSP ファイルが複数指定された場合、java ソースファイルも同一名称となるため、先に生成された java ソースファイルは、あとから生成された java ソースファイルに上書きされます。

-root オプションを指定している場合、-d オプションで指定したディレクトリ下に -root オプションで指定した web アプリケーションのルートディレクトリから JSP ファイルまでのパスが生成され、そのディレクトリ下に java ソースファイルが生成されます。

-d:non

java ソースファイルを生成しません。JSP ファイルの確認だけを実施します。

-classpath <クラスパス>

JSP ファイルから java ファイルを生成する時に必要となるクラスのクラスパスを絶対パスで指定します。次のクラスパスが必要です。

- タグライブラリの実装クラス (タグライブラリバリデータの実装クラスも含む)
- <jsp:useBean> タグで指定した JavaBeans の実装クラス

複数のパスを指定する場合は、";" で区切って指定します。

なお、-root オプションを使用し、指定したディレクトリ下の WEB-INF/classes (UNIX の場合、WEB-INF/classes/) ディレクトリ、および WEB-INF/lib (UNIX の場合、WEB-INF/lib/) ディレクトリ下に必要な class ファイル、jar ファイルを配置する場合、このオプションで指定する必要はありません。

-root <ディレクトリ>

Web アプリケーションのルートディレクトリを指定します。JSP ファイルで include ディレクティブを使用する場合、タグライブラリを使用する場合、または java ソースファイル生成時に必要なクラスパスを指定する場合に、このオプションを指定します。-root オプションを指定した場合、指定した Web アプリケーションのルートディレクトリ下に JSP ファイルを配置する必要があります。Web アプリケーションのルートディレクトリ下以外にある JSP ファイルを指定した場合は無視されます。

include ディレクティブを使用する場合

include ディレクティブの page 属性に "/" から始まるパスを指定した場合、このオプションで指定されたルートディレクトリからインクルード対象の JSP

ファイル、または静的コンテンツが検索されます。また、page 属性に "/" から始まるパスを指定していても、-root オプションを指定していない場合は、コマンドに指定した JSP ファイルのディレクトリから "/" 以下のファイルが検索されます。

タグライブラリを使用する場合

-taglib オプションの <location> に指定されたパス、または -taglib オプションを使用しないで JSP ファイル内に直接記述された tld ファイルの検索に使用されます。タグライブラリを使用する場合、-root オプションは必ず指定してください。

tld ファイルが jar ファイルに含まれる場合、その jar ファイルの検索に使用されます。この場合、使用する jar ファイルは指定した Web アプリケーションのルートディレクトリ下の WEB-INF/lib/ (UNIX の場合、WEB-INF/lib/) ディレクトリ下に配置する必要があります。

クラスパスを指定する場合

java ファイル生成時にクラスパスの指定が必要な場合、Web アプリケーションのルートディレクトリ下の WEB-INF/classes/ (UNIX の場合、WEB-INF/classes/) ディレクトリ、または WEB-INF/lib/ (UNIX の場合、WEB-INF/lib/) ディレクトリ下に class ファイル、jar ファイルを配置します。クラスパスの指定が必要な場合については、-classpath オプションの説明を参照してください。

-taglib <uri> <location>

タグライブラリを使用する場合に、<uri> および <location> に、それぞれ web.xml に定義する <taglib-uri> タグ、<taglib-location> タグの内容を指定します。-taglib オプションを指定する場合、-root オプションで Web アプリケーションのルートディレクトリを指定する必要があります。

複数のタグライブラリを使用する場合、-taglib <uri> <location> の形式で複数指定します。

<JSP ファイル名>

java ソースファイルを生成する JSP ファイルを指定します。複数の JSP ファイルを指定する場合は、空白で区切って指定します。

複数の JSP ファイルを指定した場合、同名の JSP ファイルがあるときは、先に生成された java ソースファイルは、あとから生成された java ソースファイルに上書きされます。

複数の JSP ファイルを指定した場合、指定した順に java ソースが生成されます。途中でエラーを検出した場合は java ソースファイルの生成が中断されます。

戻り値

0 :

正常終了しました。

15. Web アプリケーションで使用するコマンド

1 :

オプション解析エラーです。

2 以上 :

異常終了しました。

注意事項

このコマンドで生成された java ソースファイルは、javac コマンドでコンパイルすることで、`<%! ... %>` タグ、`<% ... %>` タグ、または `<%= ... %>` タグで記述した java 言語ソースコードのコンパイル確認もできます。ただし、生成された java ソースファイルおよび javac コマンドで作成した class ファイルは、サーバの実行環境では使用できません。

また、cjsp2java コマンドで生成したソースファイルをコンパイルするには、次に示す JAR ファイルをクラスパスに指定する必要があります。

- `<Cosminexus のインストールディレクトリ>%CC%\lib\ejbserver.jar`
- `<Cosminexus のインストールディレクトリ>%CC%\lib\hitj2ee.jar`

16 構築済み実行環境の切り替えで使用するコマンド

この章では、構築済み実行環境の切り替えで使用するコマンドの入力形式、機能などについて説明します。
このコマンドは UNIX 用です。

16.1 構築済み実行環境の切り替えで使用するコマンドの一覧

16.2 構築済み実行環境の切り替えで使用するコマンドの詳細

16.1 構築済み実行環境の切り替えで使用するコマンドの一覧

構築済み実行環境の切り替えで使用するコマンドの一覧を次の表に示します。

表 16-1 構築済み実行環境の切り替えで使用するコマンドの一覧

コマンド名称	分類	概要
cosmienv	構築済み実行環境の退避，回復，削除，一覧出力	実行環境の退避・回復を行います。

16.2 構築済み実行環境の切り替えで使用するコマンドの詳細

構築済み実行環境の切り替えで使用するコマンドの入力形式，機能などを次に示します。

cosmienv (構築済み実行環境の退避，回復，削除，一覧出力)

形式

```
cosmienv <オプション> [-I<識別子>]
                        [-P<退避パス>]
                        [-f]
```

機能

実行環境の退避・回復，退避環境の削除または一覧を表示します。

引数

<オプション>

- s
構築済み実行環境を退避します。
- d
退避環境を削除します。
- r
退避環境を回復します。
- l
現在の退避環境の一覧を表示します。
- h
cosmienv コマンドのヘルプを表示します。

-I<識別子>

- 退避環境を識別するための識別子を指定します。
- 20 文字までの半角英数字で指定します。
- s, -d, -r オプションを使用する場合には，識別子の指定は必須です。

-P<退避パス>

- 構築済み実行環境を退避するパス名をフルパスで指定します。
- 半角空白，全角空白，タブ文字を含むパス名は指定できません。
- 構築済み実行環境の退避先は <退避パス><識別子> となります。
- s オプションを使用する場合には，退避パスの指定は必須です。

16. 構築済み実行環境の切り替えで使用するコマンド

-f

- r オプションを指定した場合に、退避環境を強制的に回復することを指定します。
- f オプションを指定した場合、Cosminexus のインストール先にファイルが残っていても、退避環境の回復が実行されます。
- f オプションを指定しない場合、Cosminexus のインストール先にファイルが残っていると、退避環境の回復は実行されません。
- r オプション以外のオプションと組み合わせて指定した場合、無効です。

出力形式

退避環境の一覧表示

```
aa...aa bb...bb cccc/dd/ee ff:gg:hh ii...ii jj...jj kk...kk
```

- aa...aa 退避機能実行時に指定した識別子
- bb...bb 退避機能実行時に指定した退避パス
- cccc 退避機能実行時の時刻（年）
- dd 退避機能実行時の時刻（月）
- ee 退避機能実行時の時刻（日）
- ff 退避機能実行時の時刻（時）
- gg 退避機能実行時の時刻（分）
- hh 退避機能実行時の時刻（秒）
- ii...ii 退避した製品の形名
- jj...jj 退避した製品の VV-RR-SS
- kk...kk 退避した製品名

入力例・出力例

入力例

退避環境の一覧を表示します。

```
cosmienv -l
```

出力例

```
KEUC00001-I The cosmienv -l command will now start.  
UCAPSE001 /home/CosmiBackup 2007/03/23 15:02:36 P-1M43-7K71  
07-50 uCosminexus Application Server Enterprise  
UCAPSS001 /home/CosmiBackup 2007/03/23 16:06:08 P-1M43-7D71  
07-50 uCosminexus Application Server Standard  
KEUC00002-I The cosmienv -l command ended successfully.
```

戻り値

コマンドの戻り値を次の表に示します。

表 16-2 cosmienv コマンドの戻り値一覧

戻り値	意味
0	正常終了しました。
128	異常終了しました。
129	構築済み実行環境の退避・回復の元情報が正しくありません。
130	構築済み実行環境の退避・回復の後処理で障害が発生しました。
131	構築済み実行環境の退避・回復のための情報が正しくありません。
133	ディスクのアクセスでエラーが発生しました。
135	ファイルのオープンでエラーが発生しました。
138	引数の指定に誤りがあります。
139	製品がサポートしていない機能を指定しています。
140	コマンド実行中です。
141	コマンドの実行環境が正しくありません。
142	コマンドの実行権限がありません。
143	製品の構成ソフトウェアで障害が発生しました。 構成ソフトウェア の戻り値が付与されています。

注 障害が発生した構成ソフトウェアについては、コマンドで出力されるログファイルで確認してください。

戻り値「143」で付与される構成ソフトウェア別の戻り値を次の表に示します。

表 16-3 構成ソフトウェア別の戻り値一覧

構成ソフトウェア	戻り値	意味
Cosminexus Component Container	145	HNTRLib のインストールに失敗しました。
	146	退避に失敗しました。
	147	異常終了しました。
Cosminexus Component Transaction Monitor	129	退避・回復の元情報が正しくありません。
	130	退避・回復の後処理でエラーが発生しました。
	131	退避・回復のための情報が正しくありません。
	138	引数の指定に誤りがあります。
	142	実行権限がありません。
	146	退避先ディレクトリがありません。
	147	そのほかのエラーが発生しました。
	148	ファイルおよびディレクトリの削除に失敗しました。

16. 構築済み実行環境の切り替えで使用するコマンド

構成ソフトウェア	戻り値	意味
Cosminexus DABroker Library	128	異常終了しました。
	131	退避・回復の元情報が正しくありません。
	138	引数の指定に誤りがあります。
	144	Cosminexus DABroker Library が実行中です。
Cosminexus Developer's Kit for Java	131	構築済み実行環境の退避・回復のための情報が正しくありません。
	138	引数の指定に誤りがあります。
	144	回復に失敗しました。
	145	/opt/Cosminexus/jdk がシンボリックリンクファイルです。
	146	そのほかのエラーが発生しました。
Cosminexus Performance Tracer	129	退避・回復の元情報が正しくありません。
	130	退避・回復の後処理でエラーが発生しました。
	131	退避・回復のための情報が正しくありません。
	138	引数の指定に誤りがあります。
	142	実行権限がありません。
	145	HCCLib のインストールに失敗しました。
	146	退避先のディレクトリがありません。
	147	そのほかのエラーが発生しました。
	148	ファイルおよびディレクトリの削除に失敗しました。
Cosminexus Reliable Messaging	129	退避・回復の元情報が正しくありません。
	130	退避・回復の後処理でエラーが発生しました。
	144	HNTRLib2 のインストールに失敗しました。
	145	退避先のディレクトリがありません。
	146	そのほかのエラーが発生しました。
Cosminexus TPBroker	129	退避・回復の元情報が正しくありません。
	130	退避・回復の後処理でエラーが発生しました。
	131	退避・回復のための情報が正しくありません。
	133	ディスクのアクセスでエラーが発生しました。

構成ソフトウェア	戻り値	意味
	135	ファイルのオープンでエラーが発生しました。
	144	HCCLib のインストールに失敗しました。
	145	HNTRLib2 のインストールに失敗しました。
	146	退避に失敗しました。
	147	日立 PP インストーラ情報の取得・設定に失敗しました。
Cosminexus Web Services - Security	129	退避・回復の元情報が正しくありません。
	130	退避・回復の後処理でエラーが発生しました。
	131	退避・回復のための情報が正しくありません。
	138	引数の指定に誤りがあります。
	146	退避に失敗しました。
	147	そのほかのエラーが発生しました。
	148	ファイルおよびディレクトリの削除に失敗しました。
Cosminexus XML Processor	128	異常終了しました。
	138	引数の指定に誤りがあります。
	144	退避に失敗しました。
	145	HNTRLib2 のインストールに失敗しました。
Hitachi Web Server	128	異常終了しました。
	138	引数の指定に誤りがあります。
	144	退避に失敗しました。

注意事項

このコマンドはスーパーユーザで実行してください。

退避機能実行時は、アプリケーションサーバが提供するすべてのプログラムが停止中であることを確認してください。

回復機能実行時は、アプリケーションサーバが提供するすべての製品がアンインストール済みであることを確認してください。

すべての製品をアンインストールし、かつアプリケーションサーバのインストール先にファイルやディレクトリが残った状態で回復機能を実行したい場合は、`-f` オプションを指定してください。この場合、回復処理によってインストール先に残ったファイルが更新されることがあります。

退避機能で退避する範囲以外でアプリケーションサーバの動作に必要なものは、ユー

16. 構築済み実行環境の切り替えで使用するコマンド

ザで退避してください。退避機能で退避する範囲およびユーザで退避が必要なファイルについては、マニュアル「Cosminexus アプリケーションサーバ 機能解説 拡張編」を参照してください。

17 Web コンテナサーバで使用するコマンド

この章では、Web コンテナサーバで使用するコマンドの入力形式、機能などについて説明します。

17.1 Web コンテナサーバで使用するコマンドの一覧

17.2 Web コンテナサーバで使用するコマンドの詳細

17.1 Web コンテナサーバで使用するコマンドの一覧

Web コンテナサーバで使用するコマンドの一覧を、次の表に示します。

表 17-1 Web コンテナサーバで使用するコマンドの一覧

コマンド名称	分類	概要
cjdumpweb	Web コンテナサーバのスレッドダンプの取得	Web コンテナサーバで障害が発生した場合などに、JavaVM のスレッドダンプを取得します。
cjenvsetup	Component Container 管理者のセットアップ	インストールファイルの属性を変更して、root 権限を持つユーザ以外で運用できるようにします。 このコマンドは、UNIX 用です。 このコマンドについては、「2. J2EE サーバで使用するコマンド」の「cjenvsetup」を参照してください。
cjgetsysinfo	OS 状態情報の取得	OS の状態情報を取得します。 このコマンドについては、「2. J2EE サーバで使用するコマンド」の「cjgetsysinfo」を参照してください。
cjstartweb	Web コンテナサーバの開始	Web コンテナサーバを開始します。
cjstopweb	Web コンテナサーバの停止	Web コンテナサーバを停止します。
cjwebchangedbid	データベース ID の変更	プール管理情報のデータベース ID を変更します。
cjwebclearpool	プール内の接続のクリア	データベースの接続をクローズして、クローズした接続をプールから削除します。
cjwebclosepool	プール内の接続の強制切断	データベースの接続をプールから削除して、クローズします。
cjwebdeletepool	プール管理情報の削除	データベースのプール管理情報を削除します。
cjwebeditpool	プール管理情報の登録と変更	プール管理情報を新規登録または変更します。
cjweblistpool	プール管理情報の一覧表示	プール管理情報の一覧を表示します。
cjwebsetup	Web コンテナサーバのセットアップとアンセットアップ	Web コンテナの環境をセットアップまたはアンセットアップします。

セットアップ系コマンド (cjwebsetup) は単独で実行してください。ほかのセットアップ系コマンド (cjenvsetup, cjenvupdate, cjrarupdate, cjsetup など) と同時に実行しないでください。

17.2 Web コンテナサーバで使用するコマンドの詳細

Web コンテナサーバで使用する各コマンドの入力形式、機能などを次に示します。

コマンドの格納先

Web コンテナサーバで使用するコマンドは、次のディレクトリに格納されています。

cjdumpweb コマンド、cjstartweb コマンド、cjstopweb コマンド、cjwebsetup コマンド

- Windows の場合
`<Cosminexus のインストールディレクトリ>%CC%\web\bin%`
- UNIX の場合
`/opt/Cosminexus/CC/web/bin/`

それ以外のコマンド

- Windows の場合
`<Cosminexus のインストールディレクトリ>%CC%\admin\bin%`
- UNIX の場合
`/opt/Cosminexus/CC/admin/bin/`

cjdumpweb (Web コンテナサーバのスレッドダンプの取得)

形式

`cjdumpweb [<サーバ名称>]`

機能

Web ブラウザおよび EJB クライアントに応答が返らない場合、Web コンテナサーバで障害が発生した場合、アプリケーションの挙動を調査したい場合などに、JavaVM のスレッドダンプを取得します。

コマンドを実行すると、サーバ標準出力ログ (`cjstdout.log`), および次のファイルにスレッドダンプが出力されます。

- Windows の場合
`<Cosminexus のインストールディレクトリ>%CC%\web\containers\<サーバ名称>%javacore*.txt`
- UNIX の場合
`/opt/Cosminexus/CC/web/containers/<サーバ名称>/javacore*.txt`

引数

<サーバ名称>

17. Web コンテナサーバで使用するコマンド

Web コンテナのサーバ名称を指定します。サーバ名称には、半角英数字 (0 ~ 9, A ~ Z, a ~ z), アンダースコア (_) またはハイフン (-) が使用できます。サーバ名称の先頭の文字は英数字でなければなりません。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

注意事項

UNIX の場合、`cjdumpweb` コマンドの実行には、`root` 権限または `Component Container` 管理者の権限が必要です。

コマンドの引数にサーバ名称を指定する場合、`cjwebsetup` コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjstartweb (Web コンテナサーバの開始)

形式

```
cjstartweb [<サーバ名称>] [-security]
```

機能

サーバ名称の Web コンテナサーバを開始します。

引数

<サーバ名称>

開始する Web コンテナのサーバ名称を指定します。サーバ名称には、半角英数字 (0 ~ 9, A ~ Z, a ~ z), アンダースコア (_) またはハイフン (-) が使用できます。サーバ名称の先頭の文字は英数字でなければなりません。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-security

Web コンテナの実行時保護機能を有効にします。

戻り値

0 :

正常終了しました。

1 以上 :

異常終了しました。

ユーザ定義ファイル

Windows の場合

- <Cosminexus のインストールディレクトリ >¥CC¥web¥containers¥< サーバ名称 >¥usrconf¥usrconf.cfg
- <Cosminexus のインストールディレクトリ >¥CC¥web¥containers¥< サーバ名称 >¥usrconf¥usrconf.properties

UNIX の場合

- /opt/Cosminexus/CC/web/containers/< サーバ名称 >/usrconf/usrconf.cfg
- /opt/Cosminexus/CC/web/containers/< サーバ名称 >/usrconf/usrconf.properties

注意事項

Windows の場合、cjstartweb コマンドの実行には、Administrator 権限が必要です。

UNIX の場合、cjstartweb コマンドの実行には、root 権限（Component Container 管理者を設定していない場合）、または Component Container 管理者の権限（Component Container 管理者を設定している場合）が必要です。

コマンドの引数にサーバ名称を指定する場合、cjwebsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

cjstopweb (Web コンテナサーバの停止)

形式

```
cjstopweb [<サーバ名称>]
          [-f|-fd [<スレッドダンプ取得間隔> <スレッドダンプ取得回数>]]
```

機能

サーバ名称の Web コンテナサーバを停止します。

引数

<サーバ名称>

停止する Web コンテナサーバのサーバ名称を指定します。サーバ名称には、半角英数字（0 ~ 9, A ~ Z, a ~ z）、アンダースコア（_）またはハイフン（-）が使用できます。サーバ名称の先頭の文字は英数字でなければなりません。サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。

-f

Web コンテナサーバを強制停止します。

-fd

17. Web コンテナサーバで使用するコマンド

Web コンテナサーバの障害情報を取得したあと、Web コンテナサーバを強制停止します。

<スレッドダンプ取得間隔>

スレッドダンプを取得する間隔を 1 ~ 60 の整数 (単位: 秒) で指定します。省略した場合の初期値は 3 秒です。

<スレッドダンプ取得回数>

スレッドダンプを取得する回数を 1 ~ 60 の整数 (単位: 回) で指定します。省略した場合の初期値は 10 回です。

取得する障害情報を次に示します。

1. OS の稼働状況

Windows の場合の出力先:

<Cosminexus インストールディレクトリ>%CC%web%containers%<サーバ名称>%cjsysinfo.<出力時間>.txt

UNIX の場合の出力先:

/opt/Cosminexus/CC/web/containers/<サーバ名称>/cjsysinfo.<出力時間>.txt

<出力時間> は、ymmddHHMMSS の形式になります。

なお、出力内容については、「2. J2EE サーバで使用するコマンド」の「cjgetsysinfo (OS 状態情報の取得)」を参照してください。

2. JavaVM のスレッドダンプ

<スレッドダンプ取得間隔>、<スレッドダンプ取得回数> で指定した間隔と回数でスレッドダンプを取得します。Web コンテナサーバプロセスの状態によっては指定間隔、指定回数でスレッドダンプが出力されない場合もあります。なお、出力内容については、「cjdumpweb (Web コンテナサーバのスレッドダンプの取得)」を参照してください。

3. Web コンテナサーバのクラッシュダンプ (Windows の場合) または core ファイル (UNIX の場合)

最後のスレッドダンプ取得要求後、スレッドダンプ取得間隔時間経過後にクラッシュダンプまたは core ファイルを出力します。

クラッシュダンプの出力先:

Windows Server 2003, Windows XP と Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista ではクラッシュダンプの出力先は異なります。

・Windows Server 2003, Windows XP の場合

<ワトソン博士のクラッシュダンプ出力先> %user.dmp

注 : ワトソン博士のクラッシュダンプ作成オプションの設定が必要です。

・Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista の場合

<環境変数 CJMEMDUMP_PATH で指定したディレクトリ>

%cjmemdump.dmp

注 : 環境変数 CJMEMDUMP_PATH が指定されていないか、指定したディレ

クォトリが存在しない場合、クラッシュダンプは出力されません。また、指定したディレクトリにクラッシュダンプがすでに存在する場合、クラッシュダンプは上書きされます。

core ファイルの出力先：

```
/opt/Cosminexus/CC/web/containers/<サーバ名称>/core
```

注

Linux の場合、core.<強制停止された Web コンテナサーバのプロセス ID>

戻り値

0：

正常終了しました。

1 以上：

異常終了しました。

ユーザ定義ファイル

Windows の場合

- <Cosminexus のインストールディレクトリ>%CC%web%containers%<サーバ名称>%usrconf%usrconf.cfg
- <Cosminexus のインストールディレクトリ>%CC%web%containers%<サーバ名称>%usrconf%usrconf.properties

UNIX の場合

- /opt/Cosminexus/CC/web/containers/<サーバ名称>/usrconf/usrconf.cfg
- /opt/Cosminexus/CC/web/containers/<サーバ名称>/usrconf/usrconf.properties

注意事項

Web コンテナサーバを停止させるには、必ず `cjstopweb` コマンドを使用してください。[Ctrl+C] やタスクマネージャ (Windows の場合) から停止しないでください。

`cjstopweb` コマンドで Web コンテナサーバを停止する場合、サーブレットまたは JSP のサービスメソッドが実行中のときは、該当するメソッドが終了するまで Web コンテナサーバは停止しません。なお、`cjstopweb` コマンドは、Web コンテナサーバの終了を待たないで、すぐに終了します。

サーブレットまたは JSP のサービスメソッドが実行中の状態で `cjstopweb` コマンドを実行した場合、Web コンテナサーバを停止できないことがあります。この場合、`-f` オプションまたは `-fd` オプションを指定して、再度 `cjstopweb` コマンドを実行することで Web コンテナサーバを強制停止します。

Windows の場合、`cjstopweb` コマンドの実行には、Administrator 権限が必要です。

UNIX の場合、`cjstopweb` コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限

17. Web コンテナサーバで使用するコマンド

(Component Container 管理者を設定している場合) が必要です。

コマンドの引数にサーバ名称を指定する場合、cjwebsetup コマンドで指定したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

Windows Server 2008 x86, Windows Server 2008 x64, Windows Server 2008 R2, Windows 7 または Windows Vista で -fd オプションを指定する場合、環境変数 CJMEMDUMP_PATH は次のように設定してください。

- 環境変数 CJMEMDUMP_PATH で指定した保存先のディスクに十分な空き容量があることを確認してください。クラッシュダンプのファイルサイズは WEB コンテナサーバの実メモリ所要量以上になります。
- 環境変数 CJMEMDUMP_PATH に日本語などのマルチバイト文字を含むディレクトリを指定しないでください。クラッシュダンプの出力に失敗する場合があります。
- クラッシュダンプの出力先ディレクトリには、存在するディレクトリを指定してください。

cjwebchangedbid (データベース ID の変更)

形式

```
cjwebchangedbid [-host <ホスト名称>] [-port <ポート番号>]  
                -olddbld <データベースID> -newdbld <データベースID>
```

機能

-olddbld に指定されたデータベース ID に該当するプール管理情報のデータベース ID を、-newdbld に指定されたデータベース ID に変更します。-olddbld と -newdbld に同じデータベース ID を指定された場合、処理は実行しません。

使用中のプールのデータベース ID を変更した場合は、Web コンテナを再起動した時に変更内容が有効となります。

引数

-host <ホスト名称>

接続先のホスト名称として、コマンドからのリクエスト要求を受け付けるサブレットが動作しているホスト名称を指定します。指定を省略した場合は、localhost が仮定されます。

-port <ポート番号>

コマンドからのリクエスト要求を受け付けるサブレットが使用するポート番号として、サブレットが管理されている Web コンテナのポート番号を指定します。指定を省略した場合は、8080 が仮定されます。

-olddbld <データベース ID>

変更の対象となる現在のデータベース ID を指定します。データベース ID には、半

角英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) が使用できます。

-newdbid <データベース ID>

変更後のデータベース ID を指定します。データベース ID には、半角英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) が使用できます。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

入力例

```
cjwebchangedbid -olddbaid db_1 -newdbaid db_2
```

注意事項

引数は、コマンド名の直後に指定する必要があります。引数は、コマンド名より後ろにあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <データベース ID> -olddbaid) はできません。

cjwebclearpool (プール内のコネクションのクリア)

形式

```
cjwebclearpool [-host <ホスト名称>] [-port <ポート番号>]
               [-mode {normal|plan}] [-dbid <データベースID>]
```

機能

動作種別を選択して、プール内のすべてのコネクションまたは指定されたデータベースのコネクションをクローズします。クローズしたコネクションは、プールから削除します。

このコマンドは、データベースサーバが異常終了した場合やネットワークが切断された場合などの、システム障害時に実行するコマンドです。このコマンドでは、プールからコネクションを削除してクローズすることだけを実行します。新たなコネクションは生成しません。

引数

-host < ホスト名称 >

接続先のホスト名称として、コマンドからのリクエスト要求を受け付けるサーバ

17. Web コンテナサーバで使用するコマンド

レットが動作しているホスト名称を指定します。指定を省略した場合は、localhost が仮定されます。

`-port <ポート番号>`

コマンドからのリクエスト要求を受け付けるサブレットが使用するポート番号として、サブレットが管理されている Web コンテナのポート番号を指定します。指定を省略した場合は、8080 が仮定されます。

`-mode {normal|plan}`

使用中のコネクションに対するクリア動作の種別を指定します。

- normal (通常モード)

ユーザが使用中のコネクションはプールの外へ移して、プール内からはクリアされます。データベースとの接続はそのまま維持されますが、再利用はできません。

- plan (計画モード)

ユーザが使用中のコネクションはクリアしません。ただし、再利用はできません。

`-dbid <データベース ID>`

削除対象となるプール管理情報のデータベース ID を指定します。データベース ID には、半角英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) が使用できます。

省略すると、指定した動作種別で、すべてのプール管理情報を対象に処理が実行されます。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

入力例

```
cjwebclearpool -mode normal -dbid db_1
```

注意事項

引数は、コマンド名の直後に指定する必要があります。引数は、コマンド名より後ろにあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : <データベース ID> -dbid) やオプション名と値を非対応にすること (例 : -mode <データベース ID> -dbid normal) はできません。

cjwebclosepool (プール内のコネクションの強制切断)

形式

コネクション ID を指定する場合

```
cjwebclosepool [-host <ホスト名称>] [-port <ポート番号>]
                -dbid <データベースID> -type connectionid
                -cid <コネクションID>
```

タイムアウト時間を指定する場合

```
cjwebclosepool [-host <ホスト名称>] [-port <ポート番号>]
                -dbid <データベースID> -type timeout
                -time <タイムアウト時間>
```

機能

指定されたコネクション ID に該当するコネクション、または指定されたタイムアウト時間を超過したコネクションをプールから削除して、クローズします。

指定されたタイムアウト時間を超過したコネクションをプールから削除する場合、プールまたはデータベースに対して getConnection メソッド、SQL ステートメントなどが実行されてからの経過時間が、指定された時間を超えているコネクションが対象となります。

このコマンドは、コネクションが使用中のまま長時間放置されている場合などに使用します。コネクション ID または経過時間をキーにして削除して、プール内のコネクションが [Minimum size] に達していない場合は、新たにプールにコネクションを生成します。

引数

-host < ホスト名称 >

接続先のホスト名称として、コマンドからのリクエスト要求を受け付けるサブレットが動作しているホスト名称を指定します。指定を省略した場合は、localhost が仮定されます。

-port < ポート番号 >

コマンドからのリクエスト要求を受け付けるサブレットが使用するポート番号として、サブレットが管理されている Web コンテナのポート番号を指定します。指定を省略した場合は、8080 が仮定されます。

-dbid < データベース ID >

削除対象となるプール管理情報のデータベース ID を指定します。データベース ID には、半角英数字 (0 ~ 9 , A ~ Z , a ~ z) またはアンダースコア (_) が使用できます。

-type {connectionid|timeout}

17. Web コンテナサーバで使用するコマンド

コネクションを削除する基準の種別を指定します。

- connectionid

コネクション ID を基に、プールからコネクションを削除します。

- timeout

タイムアウト時間を基に、プールからコネクションを削除します。

-cid <コネクション ID>

削除対象となるコネクションのコネクション ID を指定します。

-time <タイムアウト時間>

コネクションを削除する際のタイムアウト時間を指定します。コネクション最終アクセス時刻から、指定時間を超過したコネクションが削除対象となります。指定は秒単位で、指定できる値は 0 指定値 2147483647 の範囲になります。

戻り値

0:

正常終了しました。

1:

異常終了しました。

2:

排他エラーによってコマンドを実行できません。

入力例

コネクション ID を指定する場合

```
cjwebclosepool -dbid db_1 -type connectionid -cid connection1
```

タイムアウト時間を指定する場合

```
cjwebclosepool -dbid db_1 -type timeout -time 36000
```

注意事項

引数は、コマンド名の直後に指定する必要があります。引数は、コマンド名より後ろにあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<データベース ID> -dbid）やオプション名と値を非対応にすること（例：-dbid timeout -type <データベース ID>）はできません。

cjwebdeletepool (プール管理情報の削除)

形式

```
cjwebdeletepool [-host <ホスト名称>] [-port <ポート番号>]  
-dbid <データベースID>
```

機能

指定されたデータベース ID に該当するプール管理情報を削除します。

使用中のプール管理情報が削除された場合、動作中のプールには影響を与えません。Web コンテナを再起動した時に有効になります。

引数

-host < ホスト名称 >

接続先のホスト名称として、コマンドからのリクエスト要求を受け付けるサブレットが動作しているホスト名称を指定します。指定を省略した場合は、localhost が仮定されます。

-port < ポート番号 >

コマンドからのリクエスト要求を受け付けるサブレットが使用するポート番号として、サブレットが管理されている Web コンテナのポート番号を指定します。指定を省略した場合は、8080 が仮定されます。

-dbid < データベース ID >

削除対象となるプール管理情報のデータベース ID を指定します。データベース ID には、半角英数字 (0 ~ 9, A ~ Z, a ~ z) またはアンダースコア (_) が使用できます。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

注意事項

引数は、コマンド名の直後に指定する必要があります。引数は、コマンド名より後ろにあれば順序は任意です。ただし、オプション名と値の順序の入れ替え (例 : < データベース ID > -dbid) はできません。

cjwebeditpool (プール管理情報の登録と変更)

形式

```
cjwebeditpool [-host <ホスト名称>] [-port <ポート番号>]
               -c <プール管理情報設定ファイルパス>
```

機能

プール管理情報を、新規登録または変更します。

登録されていないデータベース ID をプール管理情報設定ファイルに定義した場合、プール管理情報を新規登録します。すでに登録されているデータベース ID をプール管理情報設定ファイルに定義した場合、プール管理情報を変更します。

プール管理情報設定ファイルに、データベース ID、クラス名および URL 名が指定されていない場合は、プール管理情報を登録できません。

プール管理情報設定ファイルは、あらかじめエディタなどを使って作成しておく必要があります。プール管理情報設定ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（アプリケーション/リソース定義）」を参照してください。

プール管理情報の変更は、データベース ID をキーにして実行されます。登録されていないデータベース ID を指定した場合、プール管理情報が新規に作成、登録されます。

引数

-host < ホスト名称 >

接続先のホスト名称として、コマンドからのリクエスト要求を受け付けるサブレットが動作しているホスト名称を指定します。指定を省略した場合は、localhost が仮定されます。

-port < ポート番号 >

コマンドからのリクエスト要求を受け付けるサブレットが使用するポート番号として、サブレットが管理されている Web コンテナのポート番号を指定します。指定を省略した場合は、8080 が仮定されます。

-c < プール管理情報設定ファイルパス >

読み込み対象となるプール管理情報ファイル設定ファイルのパスを指定します。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

2 :

排他エラーによってコマンドを実行できません。

注意事項

引数は、コマンド名の直後に指定する必要があります。引数は、コマンド名より後ろ

にあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<プール管理情報設定ファイルパス> -c）はできません。

プール管理情報を新規に登録する場合は、プール管理情報設定ファイルに <DatabaseID> , <ClassName> , <URL> を必ず指定してください。

プール管理情報を変更する場合は、プール管理情報設定ファイルに <DatabaseID> を必ず指定してください。

プール管理情報設定ファイルの <ClassName> と <URL> は対で指定してください。

プール管理情報設定ファイルのそれぞれのタグには、次の条件を満たす値を指定してください。

- <UserCount> 0
- <MinimumSize> <MaximumSize> かつ <MaximumSize> 1
- <MinimumSize> 0
- <MaximumSize> 0
- <RetryInterval> 0
- <RetryCount> 0

新規登録または変更されたプール管理情報に対して、まだ一度もコネクション取得要求がない場合、コネクション取得要求することで新規登録または変更したプール管理情報が反映されたプールが生成されます。

すでに使用中のプール管理情報を変更した場合、コマンド実行時に <DatabaseID> , <ClassName> , <URL> 以外の項目が変更されます。<DatabaseID> , <ClassName> , <URL> については、Web コンテナを再起動してプールを再作成した時に有効になります。

新規登録時に <DatabaseID> , <ClassName> , <URL> 以外の項目を省略した場合は、デフォルトの値が設定されます。

プール管理情報設定ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（アプリケーション/リソース定義）」を参照してください。

cjweblistpool (プール管理情報の一覧表示)

形式

データベースリストを表示する場合

```
cjweblistpool [-host <ホスト名称>] [-port <ポート番号>]
               -listtype dblist [-f <ファイルパス>]
```

プールリストを表示する場合

```
cjweblistpool [-host <ホスト名称>] [-port <ポート番号>]
               -dbid <データベースID> -listtype poollist
```

17. Web コンテナサーバで使用するコマンド

`[-f <ファイルパス>]`

コネクションリストを表示する場合

```
cjweblistpool [-host <ホスト名称>] [-port <ポート番号>]
              -dbid <データベースID> -userid <ユーザID>
              -listtype connectionlist [-f <ファイルパス>]
```

すべてのプール情報を表示する場合

```
cjweblistpool [-host <ホスト名称>] [-port <ポート番号>]
              -listtype all [-f <ファイルパス>]
```

機能

プール管理情報の一覧を表示します。表示できるプール管理情報を次に示します。

データベースリスト

データベースリスト（データベース ID，データベースクラス，データベース URL，ユーザ数，コネクション最小数，コネクション最大数，リトライ待ち時間，リトライ回数，およびプールの数）を表示します。または，指定されたファイルに出力します。

プールリスト

プールリスト（ユーザ ID，コネクション取得待ちカウンタ，およびコネクションの数）と，キー項目のデータベース ID を表示します。または，指定されたファイルに出力します。

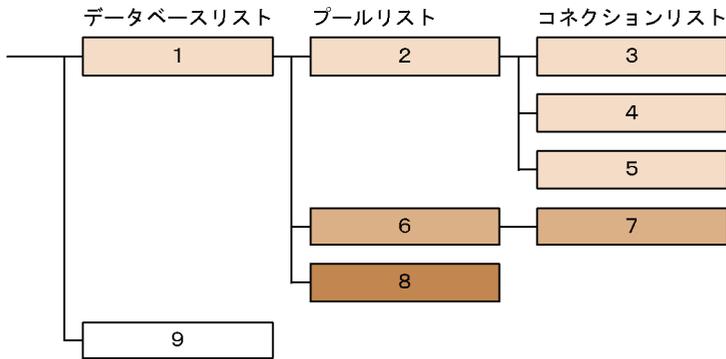
コネクションリスト

コネクションリスト（コネクション ID，使用中フラグ，再利用可否フラグ，コネクション生成時刻，コネクション最終アクセス時刻）と，キー項目のデータベース ID，ユーザ ID およびコネクション ID を表示します。または，指定されたファイルに出力します。

すべてのプール管理情報

上記のすべてのプール管理情報を一覧表示します。または，指定されたファイルに出力します。

表示順序は次の図中の番号に示した順番になります。一つのデータベースリストに関連するプールリストが表示され終わってから，次のデータベースリストが表示されます。また，一つのプールリストに関連するプールリストに関連するコネクションリストが表示され終わってから，次のプールリストが表示されます。



なお、データベースリスト、プールリストおよびコネクションリストに出力される項目については、`-listtype` オプションのオプション引数に、`dblist`、`poollist`、`connectionlist` を指定した場合と同じです。

引数

`-host` < ホスト名称 >

接続先のホスト名称として、コマンドからのリクエスト要求を受け付けるサブレットが動作しているホスト名称を指定します。指定を省略した場合は、`localhost` が仮定されます。

`-port` < ポート番号 >

コマンドからのリクエスト要求を受け付けるサブレットが使用するポート番号として、サブレットが管理されている Web コンテナのポート番号を指定します。指定を省略した場合は、`8080` が仮定されます。

`-listtype` {`dblist`|`poollist`|`connectionlist`|`all`}

表示対象となるプール管理情報の種別を指定します。

- `dblist`
データベースリストを一覧表示します。
- `poollist`
プールリストを一覧表示します。
- `connectionlist`
コネクションリストを一覧表示します。
- `all`
すべてのプール管理情報を一覧表示します。

`-f` < ファイルパス >

プール管理情報の出力先ファイルのパスを指定します。

`-dbid` < データベース ID >

処理対象となるプール管理情報のデータベース ID を指定します。データベース ID には、半角英数字 (`0 ~ 9`, `A ~ Z`, `a ~ z`) またはアンダースコア (`_`) が使用でき

17. Web コンテナサーバで使用するコマンド

ます。

-userid <ユーザ ID>

処理対象となるプール管理情報のユーザ ID を指定します。

入力例

データベースリストを表示する場合

```
cjweblispool -listtype dblist -f db1.list
```

プールリストを表示する場合

```
cjweblispool -dbid db_1 -listtype poollist -f pool1.list
```

コネクションリストを表示する場合

```
cjweblispool -dbid db_1 -userid user1 -listtype connectionlist  
-f connection1.list
```

すべてのプール情報を表示する場合

```
cjweblispool -listtype all -f all.list
```

出力形式

データベースリストを表示する場合

```
<DatabaseID=<データベースID1>><ClassName=<データベースクラス>><URL=<  
データベースURL>><UserCount=<ユーザ数>><MinimumSize=<コネクション最小  
数>><MaximumSize=<コネクション最大数>><RetryInterval=<リトライ待ち時  
間>><RetryCount=<リトライ回数>><PoolCount=<プールの数>>  
<DatabaseID=<データベースID2>><ClassName=<データベースクラス>><URL=<  
データベースURL>><UserCount=<ユーザ数>><MinimumSize=<コネクション最小  
数>><MaximumSize=<コネクション最大数>><RetryInterval=<リトライ待ち時  
間>><RetryCount=<リトライ回数>><PoolCount=<プールの数>>  
.:.  
.:.
```

プールリストを表示する場合

```
<DatabaseID=<データベースID1>><UserID=<ユーザID1>><WaitCount=<コネ  
クション取得待ちカウンタ>><ConnectionCount=<コネクションの数>>  
<DatabaseID=<データベースID1>><UserID=<ユーザID2>><WaitCount=<コネ  
クション取得待ちカウンタ>><ConnectionCount=<コネクションの数>>  
<DatabaseID=<データベースID2>><UserID=<ユーザID3>><WaitCount=<コネ  
クション取得待ちカウンタ>><ConnectionCount=<コネクションの数>>  
.:.  
.:.
```

コネクションリストを表示する場合

```
<DatabaseID=<データベースID1>><UserID=<ユーザID1>><ConnectionID=<  
コネクションID1>><ConnectionBusy=<使用中フラグ>><ConnectionReuse=<  
再利用可否フラグ>><ConnectionTime=<コネクション生成時刻>><LastTime=<  
最終アクセス時刻>>  
<DatabaseID=<データベースID1>><UserID=<ユーザID2>><ConnectionID=<  
コネクションID2>><ConnectionBusy=<使用中フラグ>><ConnectionReuse=<  
再利用可否フラグ>><ConnectionTime=<コネクション生成時刻>><LastTime=<  
最終アクセス時刻>>  
<DatabaseID=<データベースID2>><UserID=<ユーザID3>><ConnectionID=<  
コネクション
```

```

ID3>><ConnectionStatus=NotUsed><ReuseStatus=NotReuse><ConnectionTime=2003/11/25 11:36:00.616><LastTime=2003/11/25 11:40:00.313>
<DatabaseID=<データベースID2>><UserID=<ユーザID3>><ConnectionID=<コネクションID4>><ConnectionStatus=Used><ReuseStatus=Reuse><ConnectionTime=2003/11/25 11:37:00.256><LastTime=2003/11/25 11:37:00.256>
...

```

すべてのプール管理情報を表示する場合

```

<DatabaseID=<データベースID1>><ClassName=<データベースクラス>><URL=<データベースURL>><UserCount=<ユーザ数>><MinimumSize=<コネクション最小数>><MaximumSize=<コネクション最大数>><RetryInterval=<リトライ待ち時間>><RetryCount=<リトライ回数>><PoolCount=<プールの数>>
<DatabaseID=<データベースID1>><UserID=<ユーザID1>><WaitCount=<コネクション取得待ちカウンタ>><ConnectionCount=<コネクションの数>>
<DatabaseID=<データベースID1>><UserID=<ユーザID1>><ConnectionID=<コネクションID1>><ConnectionStatus=<使用中フラグ>><ReuseStatus=<再利用可否フラグ>><ConnectionTime=<コネクション生成時刻>><LastTime=<最終アクセス時刻>>
<DatabaseID=<データベースID1>><UserID=<ユーザID2>><WaitCount=<コネクション取得待ちカウンタ>><ConnectionCount=<コネクションの数>>
<DatabaseID=<データベースID1>><UserID=<ユーザID2>><ConnectionID=<コネクションID2>><ConnectionStatus=<使用中フラグ>><ReuseStatus=<再利用可否フラグ>><ConnectionTime=<コネクション生成時刻>><LastTime=<最終アクセス時刻>>
<DatabaseID=<データベースID2>><ClassName=<データベースクラス>><URL=<データベースURL>><UserCount=<ユーザ数>><MinimumSize=<コネクション最小数>><MaximumSize=<コネクション最大数>><RetryInterval=<リトライ待ち時間>><RetryCount=<リトライ回数>><PoolCount=1>
<DatabaseID=<データベースID2>><UserID=<ユーザID3>><WaitCount=<コネクション取得待ちカウンタ>><ConnectionCount=2>
<DatabaseID=<データベースID2>><UserID=<ユーザID3>><ConnectionID=<コネクションID3>><ConnectionStatus=NotUsed><ReuseStatus=NotReuse><ConnectionTime=2003/11/25 11:36:00.616><LastTime=2003/11/25 11:40:00.313>
<DatabaseID=<データベースID2>><UserID=<ユーザID3>><ConnectionID=<コネクションID4>><ConnectionStatus=Used><ReuseStatus=Reuse><ConnectionTime=2003/11/25 11:37:00.256><LastTime=2003/11/25 11:37:00.256>
...

```

戻り値

- 0 : 正常終了しました。
- 1 : 異常終了しました。
- 2 : 排他エラーによってコマンドを実行できません。

注意事項

引数は、コマンド名の直後に指定する必要があります。引数は、コマンド名より後ろにあれば順序は任意です。ただし、オプション名と値の順序の入れ替え（例：<データベース ID> -dbid）やオプション名と値を非対応にすること（例：-dbid timeout -type <データベース ID>）はできません。

cjwebsetup (Web コンテナサーバのセットアップとアンセットアップ)

形式

セットアップの場合

```
cjwebsetup [<サーバ名称>]
```

アンセットアップの場合

```
cjwebsetup -d <サーバ名称>
```

機能

指定したサーバ名称の Web コンテナの環境をセットアップまたはアンセットアップします。

-d オプションを指定した場合、次に示す Web コンテナの環境（ディレクトリおよびファイル）の <サーバ名称> 以下を削除します。

- Windows の場合
<Cosminexus のインストールディレクトリ>%CC%web%containers%<サーバ名称>%
- UNIX の場合
/opt/Cosminexus/CC/web/containers/<サーバ名称>/

引数

<サーバ名称>

セットアップまたはアンセットアップする Web コンテナのサーバ名称を指定します。

サーバ名称には、半角英数字（0～9, A～Z, a～z）、アンダースコア（_）またはハイフン（-）が使用できます。サーバ名称の先頭の文字は英数字でなければなりません。また、サーバ名称は、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の Web コンテナの作業ディレクトリの説明を参照して見積もった範囲内の長さで指定できます。

サーバ名称を省略したときは、ホスト名称がサーバ名称として使用されます。ただし、アンセットアップ時は省略できません。また、ホスト名称にサーバ名称として使用できない文字が含まれている場合はエラーになります。

-d

Web コンテナの環境を削除する場合に指定します。

戻り値

0 :

正常終了しました。

1 :

異常終了しました。

注意事項

セットアップ済みのサーバ名称を指定して再セットアップ (`cjwebsetup <サーバ名称>`) を行うと、エラーとなります。

Windows の場合、サーバ名称の大文字・小文字は区別されません。作成済みのサーバ名称と、大文字・小文字の区別以外は一致している文字列を `cjwebsetup` のサーバ名称として指定した場合、次のように動作します。

- セットアップの場合 (`cjwebsetup <サーバ名称>`)
セットアップに失敗します。
- アンセットアップの場合 (`cjwebsetup -d <サーバ名称>`)
すでに作成済みである、`<サーバ名称>` と大文字・小文字の区別以外は一致しているサーバが削除されます。

Web コンテナサーバの開始または停止コマンドを実行する場合、これらのコマンドにサーバ名称を引数として指定するときは、`cjwebsetup` コマンドで作成したサーバ名称と大文字・小文字を一致させた文字列を指定する必要があります。

起動中の Web コンテナサーバに対して、`-d` オプションで削除しないでください。

Windows の場合、削除対象のファイルの属性が読み取り専用、または使用中のときに、`-d` オプションで削除しないでください。

UNIX の場合、`cjwebsetup` コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

UNIX の場合、削除対象のディレクトリ、ファイルのアクセス権の設定および使用状態に関係なく、すべて削除します。

Web コンテナサーバを再セットアップする場合、`cjwebsetup -d <サーバ名称>` でサーバの環境をいったん削除してから再度セットアップコマンドを実行してください。

付録

付録 A	JavaVM で使用するポート番号と設定変更方法
付録 B	dabsetup (Cosminexus DABroker Library のセットアップ)
付録 C	J2EE アプリケーションを実行するシステムの構築と削除
付録 D	J2EE アプリケーションを実行するシステムの起動と停止 (Windows の場合)
付録 E	J2EE アプリケーションを実行するシステムの起動と停止 (UNIX の 場合)
付録 F	バッチアプリケーションを実行するシステムの構築と削除
付録 G	バッチアプリケーションを実行するシステムの起動と停止 (Windows の場合)
付録 H	バッチアプリケーションを実行するシステムの起動と停止 (UNIX の場合)
付録 I	Smart Composer 機能で使用できる抽象パラメタ
付録 J	このマニュアルの参考情報

付録 A JavaVM で使用するポート番号と設定変更方法

JavaVM で使用するポート番号，ポート番号の目的，およびポート番号の設定変更方法について，次の表に示します。

表 A-1 JavaVM で使用するポート番号一覧

ポート番号	目的	変更方法
1049 (TCP)	orbd コマンドが使用します。 orbd コマンドは，このポートで 持続オブジェクトに対する要求 を受け取ります。	orbd コマンドの <code>-port</code> オプションで変更でき ます。 % <code>orbd -port <ポート番号></code>
1080 (TCP)	SOCKS プロトコルで用いる ポート番号です。	プロパティ <code>socksProxyPort</code> にポート番号を指 定して変更できます。 % <code>java -DsocksProxyPort=<ポート番号></code>
1098 (TCP)	rmid コマンドが使用します。 rmid コマンドは，このポートで 内部レジストリを起動します。	rmid コマンドの <code>-port</code> オプションで変更できま す。 % <code>rmid -port <ポート番号></code>
1099 (TCP)	rmiregistry コマンドが使用しま す。rmiregistry コマンドは，こ のポート番号上にリモートオブ ジェクトレジストリを作成しま す。	rmiregistry コマンドにポート番号を指定する ことで変更できます。 % <code>rmiregistry <ポート番号></code>

注

`socks` プロパティについては，J2EE サーバ，Web コンテナサーバ，または EJB クライアントアプリケーションのシステムプロパティとして設定します。詳細については，マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（アプリケーション/リソース定義）」を参照してください。

付録 B dabsetup (Cosminexus DABroker Library のセットアップ)

ここでは、UNIX の場合に使用する Cosminexus DABroker Library のセットアップコマンドについて説明します。

なお、dabsetup コマンドを実行する前に、次の操作が必要です。

- Cosminexus DABroker Library の管理ユーザの登録
- Cosminexus DABroker Library インストールディレクトリ (/opt/DABroker) の属性 (ユーザ、グループ) の変更

Cosminexus DABroker Library のセットアップについては、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

コマンドの格納先

/opt/DABroker/bin/

(1) 形式

dabsetup [-d] [<パス名>]

(2) 機能

Cosminexus DABroker Library の運用ディレクトリを設定します。dabsetup コマンドを実行すると、インストールディレクトリのディレクトリおよびファイルが、<パス名> に指定した運用ディレクトリに移動されます。

また、-d オプションを指定することで、運用ディレクトリを削除できます。

(3) 引数

-d

Cosminexus DABroker Library 運用ディレクトリ下の Cosminexus DABroker Library システムを、OS から削除します。Cosminexus DABroker Library 運用ディレクトリを設定する場合は、このオプションを指定しないでください。

<パス名>

Cosminexus DABroker Library を運用したい任意のディレクトリを指定します。ここで指定したディレクトリ下に "DABroker" ディレクトリが作成されて、インストールディレクトリのディレクトリおよびファイルが移動されます。指定を省略した場合、Cosminexus DABroker Library 運用ディレクトリは Cosminexus DABroker Library のインストールディレクトリ (/opt/DABroker) となります。なお、リモートファイルシステム上のディレクトリ、リモートファイルシステム、またはローカルファイルとリンクしたディレクトリは指定できません。

(4) 入力例

Cosminexus DABroker Library 運用ディレクトリを設定する場合

```
dabsetup /home
```

Cosminexus DABroker Library システムを削除する場合

```
dabsetup -d /home
```

(5) 注意事項

dabsetup コマンドは、スーパーユーザで実行してください。

dabsetup コマンドは、Cosminexus DABroker Library が停止中に実行してください。

dabsetup コマンドで Cosminexus DABroker Library を再度セットアップする場合、運用ディレクトリの Cosminexus DABroker Library の動作状態は初期状態になります。ただし、ユーザが作成した Cosminexus DABroker Library 動作環境定義ファイル、ログファイル、およびトレースファイルなどはセットアップされません。

運用ディレクトリとして指定したパスが、ファイルとして存在する場合、dabsetup コマンドの実行を中止します。

運用ディレクトリとして指定したパスに、Cosminexus DABroker Library が提供するディレクトリと同一名称のファイル、または Cosminexus DABroker Library が提供するファイルと同一名称のディレクトリが存在する場合、dabsetup コマンドの実行を中止します。

運用ディレクトリとして指定したパスが、ディレクトリとして存在する場合、上書きするかどうかを確認します。なお、上書きすることを選択しても、指定したディレクトリに存在する Cosminexus DABroker Library がセットアップしないファイル (ユーザが作成した Cosminexus DABroker Library 動作環境定義ファイル、ログファイル、およびトレースファイルなど) は保証されます。

dabsetup コマンドの実行が失敗した場合、dabsetup コマンドの実行を終了して、dabsetup コマンドを実行する前の状態に戻します。

(6) Cosminexus DABroker Library 運用ディレクトリの削除について

dabsetup コマンドでの運用ディレクトリの削除に成功した場合

- インストールディレクトリに、dabsetup コマンドなどのシステムに必要なファイルが残ります。インストールディレクトリに残ったファイルは、インストーラで削除してください (アンインストールしてください)。
- ユーザが作成した Cosminexus DABroker Library 動作環境定義ファイルは、dabsetup コマンドでは削除されません。必要に応じて削除してください。

dabsetup コマンドでの運用ディレクトリの削除に失敗した場合

- 残ったディレクトリおよびファイルを OS の rm コマンドで削除してください。

付録 C J2EE アプリケーションを実行するシステムの構築と削除

ここでは、J2EE アプリケーションを実行するシステムを、ファイル編集とコマンド実行で構築、および削除する手順について説明します。

付録 C.1 システム構築の流れ

J2EE アプリケーションを実行するシステムをファイル編集とコマンド実行で構築する場合、アプリケーションサーバが提供するコマンド、ユーザ定義ファイルなどを使用してシステムの動作環境を設定します。

(1) システムの構築で使用するファイル

J2EE アプリケーションを実行するシステムを構築する場合に使用する主なファイルを次に示します。各ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編(サーバ定義)」を参照してください。

! 注意事項

ファイルを変更する場合は、J2EE サーバを停止してください。J2EE サーバの稼働中にファイルの内容を変更した場合の動作は保証されません。

usrconf.properties (J2EE サーバ用ユーザプロパティファイル)

J2EE サーバのユーザプロパティを指定するファイルです。J2EE サーバが使用するネーミングサービス、トランザクション、コネクション、コンテナ、ログ、セキュリティなどを指定します。

ファイルの格納場所を次に示します。

- Windows の場合

```
<Cosminexus のインストールディレクトリ>%CC%\server\usrconf\ejb< サーバ名称>\usrconf.properties
```

- UNIX の場合

```
/opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称>/usrconf.properties
```

usrconf.cfg (J2EE サーバ用オプション定義ファイル)

J2EE サーバの JavaVM の起動オプションを指定するファイルです。

ファイルの格納場所を次に示します。

- Windows の場合

```
<Cosminexus のインストールディレクトリ>%CC%\server\usrconf\ejb< サーバ名称>\usrconf.cfg
```

- UNIX の場合

```
/opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称>/usrconf.cfg
```

server.policy (J2EE サーバ用セキュリティポリシーファイル)

J2EE サーバを実行する JavaVM のセキュリティポリシーを指定するファイルです。ファイルの格納場所を次に示します。

- Windows の場合

<Cosminexus のインストールディレクトリ >¥CC¥server¥usrconf¥ejb¥< サーバ名称 >¥server.policy

- UNIX の場合

/opt/Cosminexus/CC/server/usrconf/ejb/< サーバ名称 >/server.policy

hitachi_web.properties (J2EE サーバ用 Web アプリケーションプロパティファイル)

Web アプリケーション個別のプロパティを指定するファイルです。なお、J2EE サーバ全体でのプロパティの設定は、usrconf.properties で指定します。

ファイルの格納場所を次に示します。

- Windows の場合

<Web アプリケーションの WEB-INF ディレクトリ >¥hitachi_web.properties

- UNIX の場合

<Web アプリケーションの WEB-INF ディレクトリ >/hitachi_web.properties

hitachi_web.properties を変更した場合、WAR ファイルを作成、インポートし直してください。

(2) J2EE アプリケーションを実行するシステムを構築する流れ

ファイル編集とコマンド実行で、J2EE アプリケーションを実行するシステムを構築する場合の、システム構築の流れについて説明します。

1. J2EE サーバをセットアップします。

デフォルトの設定のままで利用する場合は、セットアップ不要です。

インストール時にセットアップ済みの J2EE サーバとは別のサーバ名称を使用したい場合に、セットアップしてください。詳細については、「付録 C.2 J2EE サーバのセットアップ」を参照してください。

2. J2EE サーバの動作設定をします。

デフォルトの設定のままで利用する場合は、設定不要です。

使用する機能の動作を変更したり、特定の機能を使用したりする場合には、アプリケーションサーバが提供するユーザ定義ファイルやコマンドを使用して、J2EE サーバの動作を設定、変更します。

J2EE サーバの動作を設定、変更できる機能の例を次に示します。

- ネーミングサービス
- トランザクション
- JavaVM の起動オプション
- コンテナ拡張ライブラリ
- CTM によるリクエストのスケジューリング

アプリケーションサーバが提供する主な機能の設定内容については、マニュアル

「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の J2EE サーバの動作設定に関する説明を参照してください。

3. **トラブルシューティングの資料取得の設定をします。**
デフォルトの設定のまま利用する場合は、設定不要です。ただし、次の資料は、事前に取得のための設定をしておかないと、取得できません。これらの資料はトラブルシューティングで必要となるため、取得することをお勧めします。
 - OS の統計情報，ユーザダンプ（Windows の場合）または core ダンプ（UNIX の場合）
 - JavaVM のガーベージコレクションのログこれらの資料を取得する場合や、ほかの資料の取得設定を変更したい場合に、トラブルシューティングの資料取得の設定をしてください。トラブルシューティングに必要な資料の詳細と、システム構築時の設定の要否については「付録 C.1(3) トラブルシューティングに必要な資料」を参照してください。
4. **Web サーバとの連携の設定をします。**
Web サーバと連携しない場合は、設定不要です。
リダイレクタを利用したリクエストの振り分け、通信タイムアウト、エラーページのカスタマイズなど、Web サーバと連携するための設定をします。設定方法の詳細については、マニュアル「Cosminexus アプリケーションサーバ機能解説 基本・開発編（Web コンテナ）」の Web サーバとの連携の設定（アプリケーションの実行環境の場合）に関する説明を参照してください。
5. **統合ユーザ管理機能の設定をします。**
統合ユーザ管理機能を使用しない場合は、設定不要です。
Cosminexus システムにログインするユーザを統合管理するための設定をします。
ユーザ認証情報を格納する LDAP ディレクトリサーバや、統合ユーザ管理のコンフィグレーションファイルを設定します。設定方法の詳細については、マニュアル「Cosminexus アプリケーションサーバ機能解説 拡張編」の統合ユーザ管理機能の設定に関する説明を参照してください。
6. **システムの可用性を高める設定をします。**
システムの可用性を高める機能を使用しない場合は、設定不要です。
セッションフェイルオーバー機能による J2EE サーバへのセッション情報の引き継ぎの設定をします。設定方法の詳細については、マニュアル「Cosminexus アプリケーションサーバ機能解説 拡張編」の J2EE サーバ間のセッション情報の引き継ぎの設定に関する説明を参照してください。
7. **リソースと J2EE アプリケーションの設定をします。**
サーバ管理コマンドを使用して、リソースアダプタや J2EE アプリケーションのプロパティを定義し、J2EE サーバにインポートします。設定方法の詳細については、マニュアル「Cosminexus アプリケーションサーバ機能解説 基本・開発編（コンテナ共通機能）」のリソースと J2EE アプリケーションの設定に関する説明を参照してください。

8. システムを起動して、システムの動作を確認します。

システムの起動および停止については、「付録 D J2EE アプリケーションを実行するシステムの起動と停止 (Windows の場合)」, または「付録 E J2EE アプリケーションを実行するシステムの起動と停止 (UNIX の場合)」を参照してください。

! 注意事項

各ファイルでデフォルトの設定を変更する場合、簡易構築定義ファイルを使用して設定しているときは、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の J2EE サーバの動作設定についての説明を参照してください。

(3) トラブルシューティングに必要な資料

トラブルシューティングに必要な資料と、システム構築時の設定の要否について次の表に示します。なお、設定方法の詳細については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」のトラブルシューティングの資料取得の設定に関する説明を参照してください。

表 C-1 トラブルシューティングの資料の種類と設定要否 (J2EE アプリケーションを実行するシステムの場合)

資料の種類	設定の要否
snapshot ログ	1
性能解析トレースファイル	x 2
J2EE サーバのログ	
SFO サーバのログ	
Web サーバのログ	
リダイレクタのログ	
J2EE アプリケーションのユーザログ	
インプロセス HTTP サーバのログ	
稼働情報ファイル	
リソースアダプタのログ	
Cosminexus TPBroker のトレースファイル	
Cosminexus DABroker Library のログ	
OS の統計情報	
ユーザダンプ	
core ダンプ	
JavaVM の資料	
サーバ管理コマンドのログ	

(凡例)

- : 資料を取得する場合には、設定が必要。
- : デフォルトの資料取得の設定を変更する場合だけ、設定が必要。
- × : 設定は不要。

注 1

snapshotlog コマンドを使用して snapshot ログを取得します。この場合、あらかじめ設定しておいたリストファイルを、コマンド実行時に引数として指定します。snapshotlog コマンド、およびデフォルトの設定で収集できる snapshot ログについては、「11.2 ログの運用で使用するコマンドの詳細」の「snapshotlog (snapshot ログの収集)」を参照してください。なお、snapshot ログの収集先を変更する場合には、snapshot ログ収集対象定義ファイルを編集します。snapshot ログ収集対象定義ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」を参照してください。

注 2

日常的なシステム運用の作業で、cprfstart コマンドを実行してパフォーマンストレーサを起動するときに、PRF トレースの取得レベルやファイルの面数を指定できます。性能解析トレースファイルの取得については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

表 C-1 に示したログは、snapshot ログで一括収集できます。ただし、Cosminexus TPBroker のトレースファイルについては、収集できるものと収集できないものが混在しています。このほかにも、snapshot ログのデフォルトの設定で取得できないログについては、取得のための設定をしたり、snapshot ログの収集先に追加したりする必要があります。

なお、次のログはログ出力先を変更できません。

- セットアップ時に作成される install.log、移行コマンドなどのログファイル
- 次のディレクトリに出力される Java のスレッドダンプファイル

Windows の場合

< 作業ディレクトリ >%ejb%< サーバ名称 >

UNIX の場合

< 作業ディレクトリ >/ejb/< サーバ名称 >

ログの種類やデフォルト値、チャンネル名、取得できるログの詳細や取得方法については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

付録 C.2 J2EE サーバのセットアップ

アプリケーションサーバのシステムを構成する J2EE サーバのセットアップについて説明します。

J2EE サーバは、製品のインストール時にセットアップ済みです。デフォルトの設定では、J2EE サーバ名はホスト名です。別のサーバ名に変更したい場合には、J2EE サーバをセットアップしてください。また、同じホスト内で複数の J2EE サーバを起動する場

合には、それぞれの J2EE サーバに別のサーバ名を割り当てる必要があるため、J2EE サーバをセットアップしてください。

J2EE サーバのセットアップの手順について説明します。セットアップに関する注意事項については、「付録 C.2 (3) セットアップに関する注意事項」を参照して、必要に応じて対応してください。同じホスト内で複数の J2EE サーバを起動する場合に、サーバ名を割り当てるときの注意事項については、「付録 C.2 (4) 複数の J2EE サーバを起動する場合の注意事項」を参照してください。

(1) Windows の場合

Windows の場合の J2EE サーバのセットアップ手順を次に示します。

1. 次のコマンドでカレントディレクトリを移動します。
"cd <Cosminexus のインストールディレクトリ>%CC%server%bin"
2. cjsetup コマンドで J2EE サーバをセットアップします。
"cjsetup <サーバ名>"

J2EE サーバのセットアップの例を次に示します。この例では、Cosminexus のインストールディレクトリは「C:\Program Files\Hitachi\Cosminexus」、サーバ名は「MyServer」です。

```
C:\>cd C:\Program Files\Hitachi\Cosminexus\CC\server\binC:\Program Files\Hitachi\Cosminexus\CC\server\bin>cjsetup MyServer
```

(2) UNIX の場合

UNIX の場合の J2EE サーバのセットアップ手順を次に示します。

1. cjsetup コマンドで J2EE サーバをセットアップします。コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
/opt/Cosminexus/CC/server/bin/cjsetup <サーバ名>
```

J2EE サーバのセットアップの例を次に示します。この例では、サーバ名は「MyServer」です。

```
/opt/Cosminexus/CC/server/bin/cjsetup MyServer
```

(3) セットアップに関する注意事項

J2EE サーバのセットアップは、1 回だけ実行してください。セットアップ済みの J2EE サーバを設定してセットアップすると、エラーメッセージが出力されます。

J2EE サーバをセットアップして、カスタマイズしない場合、J2EE サーバはホスト名「localhost」、ポート番号「900」の CORBA ネーミングサービスをインプロセスで自動起動して使用します。使用する CORBA ネーミングサービスを変更する場合は、usrconf.properties の ejbserver.naming.host キーと ejbserver.naming.port キーで変更してください。また、CORBA ネーミングサービスの起動モードを変更する場合には、usrconf.properties の ejbserver.naming.startupMode キーで変更してください。

J2EE サーバを開始する前に、環境変数「OSAGENT_PORT」の値とusrconf.properties の vbroker.agent.port キーの値が一致しているかどうかを確認してください。

サーバ名称に使用できる文字については、cjsetup コマンドの説明を参照してください。cjsetup コマンドについては、「2.2 J2EE サーバを操作するコマンド」を参照してください。

サーバ名称は、作業ディレクトリ中のディレクトリ名として使用されます。作業ディレクトリ下のパス長がプラットフォームの上限に達しないようにサーバ名称を指定してください。作業ディレクトリのパス長の見積もりについては、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の J2EE サーバの作業ディレクトリに関する説明を参照してください。

(4) 複数の J2EE サーバを起動する場合の注意事項

J2EE サーバはサーバ名称によって識別されます。インストール時に、インストールされたホストのホスト名がデフォルトのサーバ名称として設定されています。このため、同じホスト内で複数の J2EE サーバを起動する場合、それぞれの J2EE サーバにサーバ名称を割り当て、J2EE サーバをセットアップします。

J2EE サーバのサーバ名称を割り当てる上で、次に示す二つの条件を考慮する必要があります。

同じホスト内では、同じサーバ名称の J2EE サーバを複数回セットアップしてはいけません。同じサーバ名称で再セットアップした場合、エラーになります。

同じ CORBA ネーミングサービスを利用する場合、複数のホスト間でも同じサーバ名称の J2EE サーバを複数回セットアップしてはいけません。

J2EE サーバと接続する CORBA ネーミングサービスのホスト名およびポート番号、CORBA ネーミングサービスの使用するスマートエージェントの設定に関する注意事項を次に示します。

ホスト名およびポート番号の設定

同一ホストで複数の J2EE サーバを同時に起動する場合、CORBA ネーミングサービスとの接続に使用するホスト名およびポート番号が両方とも重ならないようにしてください。ホスト名およびポート番号は、usrconf.properties で設定します。

また、usrconf.properties の webserver.connector.ajp13.port キーおよび

`ejbserver.http.port` キーで設定されているポート番号には、同時に起動する J2EE サーバ間で異なる値を使用するように設定してください。

スマートエージェントの設定

J2EE サーバが接続する CORBA ネーミングサービスでスマートエージェントを使用する場合、同一のスマートエージェントの範囲内（同一のポート番号で接続する範囲内）で J2EE サーバ名を一意としてください。J2EE サーバ名を一意にしない場合には、サーバ管理コマンドで `KDJE37302-E` のメッセージが出力され、サーバ管理コマンドが実行できなくなることがあります。

なお、CORBA ネーミングサービスでスマートエージェントを使用する場合とは、次のとおりです。

CORBA ネーミングサービスをアウトプロセスで手動起動する場合

`ejbserver.naming.startupMode` キーに `manual` を指定して起動する場合、CORBA ネーミングサービス起動時の引数で、
`vbroker.agent.enableLocator=true`（デフォルトは `true`）を指定します。

CORBA ネーミングサービスをアウトプロセスで自動起動、またはインプロセスで自動起動する場合

`ejbserver.naming.startupMode` キーに `automatic` または `inprocess` を指定して起動する場合、`usrconf.properties` で、`vbroker.agent.enableLocator=true`（デフォルトは `false`）を指定して、スマートエージェントの使用を有効にします。

付録 C.3 システムの削除の流れ

J2EE アプリケーションを実行するシステムをファイル編集とコマンド実行で削除する場合、アプリケーションサーバが提供するコマンドを使用して、J2EE サーバをアンセットアップし、システムを削除します。

ファイル編集とコマンド実行で、J2EE アプリケーションを実行するシステムを削除する場合の、システム削除の流れについて説明します。

1. システムを停止します。

システムの停止手順および停止方法については、「付録 D J2EE アプリケーションを実行するシステムの起動と停止（Windows の場合）」、または「付録 E J2EE アプリケーションを実行するシステムの起動と停止（UNIX の場合）」を参照してください。

2. J2EE サーバをアンセットアップします。

詳細については、「付録 C.4 J2EE サーバのアンセットアップ」を参照してください。

付録 C.4 J2EE サーバのアンセットアップ

アプリケーションサーバのシステムを構成する J2EE サーバのアンセットアップについて説明します。J2EE サーバのアンセットアップの手順について、OS ごとに説明しま

す。

(1) Windows の場合

Windows の場合の J2EE サーバのアンセットアップ手順を次に示します。

1. 次のコマンドでカレントディレクトリを移動します。
"cd <Cosminexus のインストールディレクトリ>%CC%server%bin"
2. cjsetup コマンドで引数に -d を指定して、J2EE サーバをアンセットアップします。
"cjsetup -d <サーバ名>"
J2EE サーバのアンセットアップの例を次に示します。この例では、Cosminexus のインストールディレクトリは「C:%Program Files%Hitachi%Cosminexus」、サーバ名は「MyServer」です。

```
C:>cd C:\Program Files\Hitachi\Cosminexus\CC\server\bin
C:\Program Files\Hitachi\Cosminexus\CC\server\bin>cjsetup -d MyServer
```

(2) UNIX の場合

UNIX の場合の J2EE サーバのアンセットアップ手順を次に示します。

1. cjsetup コマンドで引数に -d を指定して、J2EE サーバをアンセットアップします。
コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。
/opt/Cosminexus/CC/server/bin/cjsetup -d <サーバ名>
J2EE サーバのアンセットアップの例を次に示します。この例では、サーバ名は「MyServer」です。

```
/opt/Cosminexus/CC/server/bin/cjsetup -d MyServer
```

付録 D J2EE アプリケーションを実行するシステムの起動と停止 (Windows の場合)

ここでは、Windows の場合の J2EE アプリケーションを実行するシステムの起動と停止の手順について説明します。

付録 D.1 システムの起動手順

システムの起動手順について説明します。

! 注意事項

システム構成によって、アプリケーションサーバだけを起動する場合と、アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを起動する場合があります。

アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを起動する場合、次の点に注意してください。

- セッションフェイルオーバーサーバを使用するときは、アプリケーションサーバより先にセッションフェイルオーバーサーバを起動してください。
- 統合ネーミングスケジューラサーバの起動順序は任意です。

次に、アプリケーションサーバ、統合ネーミングスケジューラサーバおよびセッションフェイルオーバーサーバの起動手順について説明します。

なお、データベース、LDAP ディレクトリサーバ、EIS などの関連製品は起動済みであることを前提に説明します。データベース、LDAP ディレクトリサーバ、EIS などを使用している場合は、必要に応じてあらかじめ起動しておいてください。

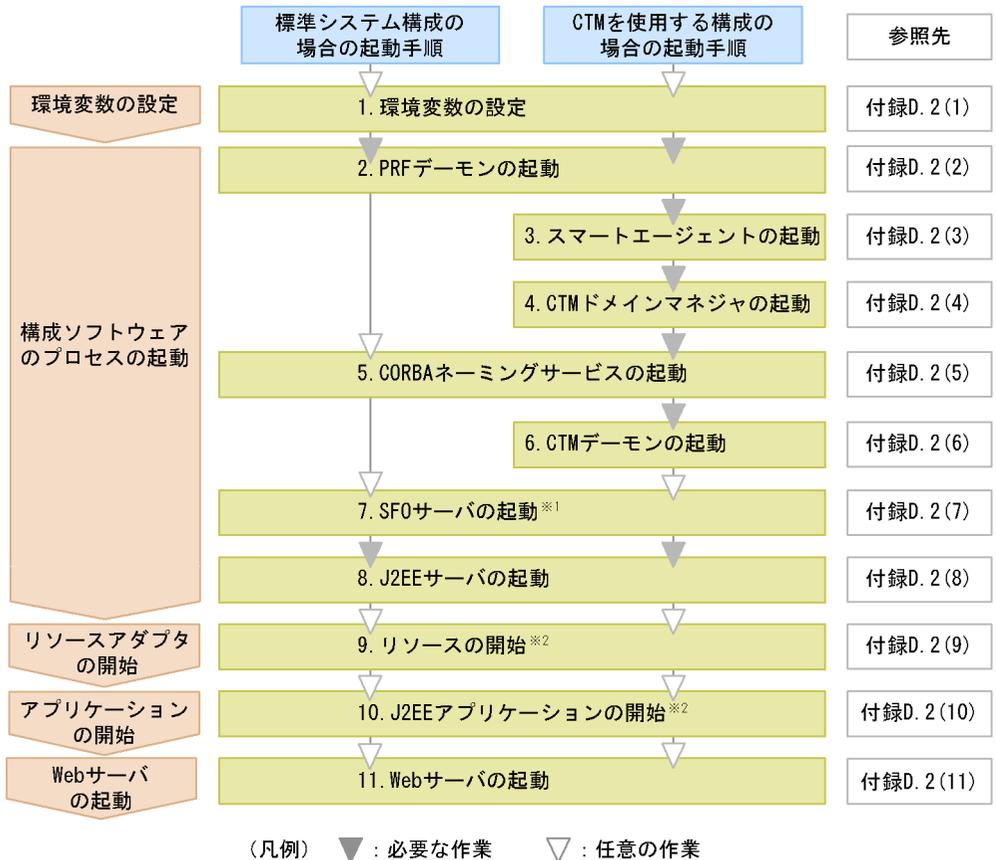
(1) アプリケーションサーバの起動手順の流れ

アプリケーションサーバの起動手順の流れについて説明します。

ここでは、標準システム構成の場合と、CTM を使用する構成の場合のアプリケーションサーバの起動手順の流れについて説明します。

アプリケーションサーバの起動手順の流れを次の図に示します。

図 D-1 アプリケーションサーバの起動の流れ (Windows の場合)



注※1 SF0サーバを起動したときに、SF0サーバアプリケーションが停止している場合、SF0サーバアプリケーションを開始します。

注※2 J2EEサーバを起動したときに、リソースまたはJ2EEアプリケーションが停止している場合だけ必要な手順です。

図中の 1. ~ 11. について説明します。

1. 環境変数の設定

環境変数を設定します。環境変数の設定方法については、「付録 D.2(1) システム起動時の環境変数の設定」を参照してください。

2. PRF デーモンの起動

cpfstart コマンドで PRF デーモンを起動します。起動方法については、「付録 D.2(2) PRF デーモンの起動」を参照してください。

3. スマートエージェントの起動

CTM を使用しない場合、この操作は不要です。

osagent コマンドでスマートエージェントを起動します。起動方法については、「付録

D.2(3) スマートエージェントの起動」を参照してください。

4. CTM ドメインマネージャの起動

CTM を使用しない場合、この操作は不要です。

ctmdmstart コマンドで CTM ドメインマネージャを起動します。起動方法については、「付録 D.2(4) CTM ドメインマネージャの起動」を参照してください。

5. CORBA ネーミングサービスの起動

J2EE サーバ用の CORBA ネーミングサービスは、J2EE サーバ起動時にインプロセスで起動することを推奨します。

アウトプロセスでの起動方法については、「付録 D.2(5) CORBA ネーミングサービスの起動」を参照してください。

6. CTM デーモンの起動

CTM を使用しない場合、この操作は不要です。

ctmstart コマンドで CTM デーモンを起動します。起動方法については、「付録 D.2(6) CTM デーモンの起動」を参照してください。

7. SFO サーバの起動

SFO サーバを起動しない場合、この操作は不要です。また、アプリケーションサーバ内で SFO サーバを起動しない場合もこの操作は不要です。

cjstartsv コマンドに `-nosecurity` オプションを指定して、SFO サーバを起動します。また、SFO サーバを起動したときに SFO サーバアプリケーションが停止している場合、サーバ管理コマンド (`cjstartapp`) で SFO サーバアプリケーションを開始します。

起動方法およびアプリケーションの開始については、「付録 D.2(7) SFO サーバの起動」を参照してください。

8. J2EE サーバの起動

cjstartsv コマンドで J2EE サーバを起動します。起動方法については、「付録 D.2(8) J2EE サーバの起動」を参照してください。

9. リソースの開始

リソースを使用しない場合、この操作は不要です。また、J2EE アプリケーションに含まれるリソースアダプタを利用する場合もこの操作は不要です。

サーバ管理コマンド (`cjstartrar`, `cjstartjb`) でリソースを開始します。開始方法については、「付録 D.2(9) リソースの開始」を参照してください。

10. J2EE アプリケーションの開始

サーバ管理コマンド (`cjstartapp`) で J2EE アプリケーションを開始します。なお、リソースアダプタを含んだ J2EE アプリケーションの場合、J2EE アプリケーションに含まれるすべてのリソースアダプタが同時に開始されます。

開始方法については、「付録 D.2(10) J2EE アプリケーションの開始」を参照してください。

11. Web サーバの起動

起動方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの起動のタイミングはシステムの運用形態に依存しますが、この順序で起動することで J2EE アプリケーションへのリクエストの受け付けを制御できます。Web サーバ起動後、Web クライアントからの J2EE アプリケーションへのリクエストの受け付けを開始します。開始方法については、「付録 D.2(11) Web サーバの起動」を参照してください。

また、インプロセス HTTP サーバを使用する場合は、Web サーバの起動は必要ありません。

(2) 統合ネーミングスケジューラサーバの起動の流れ

統合ネーミングスケジューラサーバの起動の流れを次の図に示します。

図 D-2 統合ネーミングスケジューラサーバの起動の流れ (Windows の場合)



図中の 1. ~ 6. について説明します。

1. 環境変数の設定

環境変数を設定します。環境変数の設定方法については、「付録 D.2(1) システム起動時の環境変数の設定」を参照してください。

2. PRF デーモンの起動

cprfstart コマンドで PRF デーモンを起動します。起動方法については、「付録 D.2(2) PRF デーモンの起動」を参照してください。

3. スマートエージェントの起動

osagent コマンドでスマートエージェントを起動します。起動方法については、「付録 D.2(3) スマートエージェントの起動」を参照してください。

4. CTM ドメインマネージャの起動

ctmdmstart コマンドで CTM ドメインマネージャを起動します。起動方法については、「付録 D.2(4) CTM ドメインマネージャの起動」を参照してください。

5. CORBA ネーミングサービスの起動

CTM 用のグローバル CORBA ネーミングサービスを起動します。起動方法については、「付録 D.2(5) CORBA ネーミングサービスの起動」を参照してください。

6. CTM デーモンの起動

ctmstart コマンドで CTM デーモンを起動します。起動方法については、「付録 D.2(6) CTM デーモンの起動」を参照してください。

(3) セッションフェイルオーバーサーバの起動の流れ

セッションフェイルオーバーサーバの起動の流れを次の図に示します。

図 D-3 セッションフェイルオーバーサーバの起動の流れ (Windows の場合)



注※ SF0サーバを起動したときに、SF0サーバアプリケーションが停止している場合、SF0サーバアプリケーションを開始します。

図中の 1. ~ 3. について説明します。

1. 環境変数の設定

環境変数を設定します。環境変数の設定方法については、「付録 D.2(1) システム起動時の環境変数の設定」を参照してください。

2. PRF デーモンの起動

cprfststart コマンドで PRF デーモンを起動します。起動方法については、「付録 D.2(2) PRF デーモンの起動」を参照してください。

3. SFO サーバの起動

cjstartsv コマンドに `-nosecurity` オプションを指定して、SFO サーバを起動します。また、SFO サーバを起動したときに、SFO サーバアプリケーションが停止している場合、サーバ管理コマンド (`cjstartapp`) で SFO サーバアプリケーションを開始します。

なお、SFO サーバでは、明示管理ヒープ機能の設定を無効にすることを推奨します。

起動方法およびアプリケーションの開始については、「付録 D.2(7) SFO サーバの起動」を参照してください。

！ 注意事項

SFO サーバで実行できるアプリケーションは、SFO サーバアプリケーションだけです。
SFO サーバで SFO サーバアプリケーション以外のアプリケーションは実行しないでください。

付録 D.2 システムの起動方法

ここでは、構成ソフトウェアの各プロセスの起動方法、リソースの開始方法、および J2EE アプリケーションの開始方法について説明します。

(1) システム起動時の環境変数の設定

ここでは、アプリケーションサーバ、統合ネーミングスケジューラサーバ、および SFO サーバに設定する環境変数について説明します。

set コマンドで次に示す環境変数を設定してください。

```
set PATH=<Cosminexusのインストールディレクトリ>%jdk%bin;%PATH%
set PATH=<Cosminexusのインストールディレクトリ>%TPB%bin;%PATH%
set VBROKER_ADM=<Cosminexusのインストールディレクトリ>%TPB%adm
set TPDIR=<Cosminexusのインストールディレクトリ>%TPB
set CTMDIR=<Cosminexusのインストールディレクトリ>%CTM
set CTMSPOOL=<Cosminexusのインストールディレクトリ>%CTM%spool
```

注

CTM を使用する場合に、アプリケーションサーバおよび統合ネーミングスケジューラサーバで設定が必要な環境変数です。

- 環境変数「TZ」に「JST-9」を設定して、システム環境変数に登録してください。
- CORBA ネーミングサービスを手動起動する場合は、環境変数「PATH」の先頭に「<Cosminexusのインストールディレクトリ>%jdk%bin」を指定してください。
- HiRDB または Oracle を利用する場合には、それぞれのライブラリのパスを環境変数「PATH」に設定します。あらかじめシステム環境変数に設定している場合は set コマンドの実行は不要です。
- インストール時に、「<システムドライブ>%Common Files%Hitachi」が環境変数「PATH」に追加され、自動的にシステム環境変数に登録されます。
- インストール時に、「<Cosminexusのインストールディレクトリ>%PRF%spool」が環境変数「PRFSPOOL」に設定され、自動的にシステム環境変数に登録されます。

(2) PRF デーモンの起動

PRF デーモンの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%PRF%\bin"
```

2. cprfstart コマンドで PRF デーモンを起動します。

```
cprfstart
```

(3) スマートエージェントの起動

スマートエージェントの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%TPB%\bin"
```

2. osagent コマンドでスマートエージェントを起動します。
コマンドの実行には、Administrator 権限が必要です。

```
osagent
```

(4) CTM ドメインマネージャの起動

CTM ドメインマネージャの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CTM%\bin"
```

2. ctmdmstart コマンドで CTM ドメインマネージャを起動します。
コマンドの実行には、Administrator 権限が必要です。

```
ctmdmstart
```

なお、CTM ドメインマネージャを起動する時には、パフォーマンストレーサのライブラリパス (<Cosminexus のインストールディレクトリ>%PRF%\lib) を同時に設定しないでください。パフォーマンストレーサのライブラリパスを同時に設定すると、CTM ドメインマネージャがパフォーマンストレーサのライブラリをロードした場合にエラーとなり、起動できません。

(5) CORBA ネーミングサービスの起動

CORBA ネーミングサービスの起動方法について説明します。

J2EE サーバ用の CORBA ネーミングサービスは、J2EE サーバ起動時にインプロセスで起動することを推奨します。アウトプロセスで起動する場合は、nameserv コマンドを使用して起動します。

ここでは、nameserv コマンドで CORBA ネーミングサービスをアウトプロセスで起動する方法について説明します。J2EE サーバの起動時に CORBA ネーミングサービスをインプロセスで起動する方法については、「付録 D.2(8) J2EE サーバの起動」を参照してください。

なお、CORBA ネーミングサービス起動時は次の点に留意してください。

- CTM を使用する場合、J2EE サーバ用および CTM 用の CORBA ネーミングサービスを起動します。CTM 用にグローバル CORBA ネーミングサービスを nameserv コマンドで手動起動して、さらに J2EE サーバ用の CORBA ネーミングサービスを J2EE サーバ起動時にインプロセスで自動起動してください。
- J2EE サーバ間でのセッション情報の引き継ぎを行う場合、アプリケーションサーバ内で SFO サーバを起動するときは、SFO サーバ用の CORBA ネーミングサービスはインプロセスで起動することが前提のため起動は不要です。

nameserv コマンドで CORBA ネーミングサービスを手動起動する場合の手順を次に示します。

1. set コマンドで環境変数「CLASSPATH」を設定解除します。

```
set CLASSPATH=
```

2. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%TPB%bin"
```

3. nameserv コマンドでローカル CORBA ネーミングサービスを手動開始します。
コマンドの実行には、Administrator 権限が必要です。

```
nameserv -J-Dvbroker.agent.enableLocator=false
-J-Djava.security.policy=="<Cosminexusのインストールディレクトリ>%
CC%server%sysconf%cli.policy"
-J-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<ポート番号>
```

注意事項

- CORBA ネーミングサービスを手動起動する場合は、環境変数「PATH」の先頭に「<Cosminexus のインストールディレクトリ>%jdk%bin」を指定してください。
- nameserv コマンドで指定するオプションの値に空白を含むディレクトリが含まれる場合は、値全体をダブルクォーテーション (") で囲むようにしてください。
- nameserv コマンドで指定する値は環境に合わせて変更してください。

- CORBA ネーミングサービスは、次のオプションで指定したポート番号で起動します。
-J-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<ポート番号>

(6) CTM デーモンの起動

CTM デーモンの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CTM%bin"
```

2. ctmstart コマンドで CTM デーモンを起動します。
コマンドの実行には、Administrator 権限が必要です。

```
ctmstart -CTMINSRef <ホスト名:ポート番号>
```

ホスト名およびポート番号にはグローバル CORBA ネーミングサービスを起動しているホスト名およびポート番号を指定してください。

なお、CTM デーモンを起動する時には、パフォーマンストレーサのライブラリパス (<Cosminexus のインストールディレクトリ>%PRF%lib) を同時に設定しないでください。パフォーマンストレーサのライブラリパスを同時に設定すると、CTM デーモンがパフォーマンストレーサのライブラリをロードした場合にエラーとなり、起動できません。

参考

CTM デーモンを開始する ctmstart コマンドでは、コマンド実行時に次のファイルを指定できます。

- CTM ユーザ環境変数定義ファイル
-CTMEnviron オプションで指定するファイルです。CTM デーモンで管理するプロセスが使用する環境変数を指定したい場合は、このファイルに環境変数を記述します。
- CTM コマンドオプションファイル
-CTMRegOption で指定するファイルです。CTM レギュレータを自動開始するときのコマンドオプションを指定したい場合は、このファイルにコマンドオプションを記述します。

必要に応じて、これらのファイルを作成しておいてください。

(7) SFO サーバの起動

SFO サーバの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%server%bin"
```

2. cjstartsv コマンドに -nosecurity オプションを指定して、SFO サーバを開始します。
コマンドの実行には、Administrator 権限が必要です。

```
cjstartsv <サーバ名称> -nosecurity
```

3. SFO サーバアプリケーションを開始します。
SFO サーバを起動したときに SFO サーバアプリケーションが停止している場合、サーバ管理コマンドを使用して、SFO サーバアプリケーションを開始します。なお、サーバ管理コマンドの実行には、Administrator 権限が必要です。
SFO サーバアプリケーションの名称は「CCC_SFO_Server_Application」です。

! 注意事項

SFO サーバで実行できるアプリケーションは、SFO サーバアプリケーションだけです。
SFO サーバで SFO サーバアプリケーション以外のアプリケーションは実行しないでください。

(8) J2EE サーバの起動

J2EE サーバの起動方法について説明します。

J2EE サーバ起動時に、次に示すプロセスをインプロセスで起動することを推奨します。

- J2EE サーバ用の CORBA ネーミングサービス
- トランザクションサービス (トランザクションサービスを使用する場合)

なお、CTM を使用する場合、J2EE サーバ用の CORBA ネーミングサービスはインプロセスで起動してください。

次に示す場合の J2EE サーバの起動手順について (a) ~ (d) で説明します。

- 標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合
- CTM を使用する構成で CORBA ネーミングサービスおよびトランザクションサービスをインプロセスで起動する場合
- 標準システム構成でインプロセス HTTP サーバを起動して CORBA ネーミングサービスをインプロセスで起動する場合
- 標準システム構成で CORBA ネーミングサービスをアウトプロセスで起動する場合

また、J2EE サーバ起動時は次の点に留意してください。

- 複数の J2EE サーバで一つのネーミングサービスを共有する場合、J2EE サーバは一つずつ起動してください。複数の J2EE サーバを同時に起動しようとすると J2EE

サーバの起動に失敗することがあります。

(a) 標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合

標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合の、J2EE サーバの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%\server\bin"
```

2. cjstartsv コマンドで J2EE サーバを開始します。

コマンドの実行には、Administrator 権限が必要です。

```
cjstartsv <サーバ名称>
```

(b) CTM を使用する構成で CORBA ネーミングサービスおよびトランザクションサービスをインプロセスで起動する場合

CTM を使用する構成で、J2EE サーバ用の CORBA ネーミングサービス、およびトランザクションサービスをインプロセスで起動する場合の、J2EE サーバの起動手順について説明します。

なお、J2EE サーバ用の CORBA ネーミングサービス、およびトランザクションサービスをインプロセスで起動する場合の前提を次に示します。

- usrconf.properties ファイルの ejbserver.naming.port キーに J2EE サーバ用の CORBA ネーミングサービスの TCP ポート番号が設定されている必要があります。
ejbserver.naming.port = <TCP ポート番号 >

J2EE サーバの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%\server\bin"
```

2. cjstartsv コマンドで J2EE サーバを開始します。

コマンドの実行には、Administrator 権限が必要です。

```
cjstartsv <サーバ名称>
```

(c) 標準システム構成でインプロセス HTTP サーバを起動して CORBA ネーミングサービスをインプロセスで起動する場合

標準システム構成でインプロセス HTTP サーバを起動して、CORBA ネーミングサービ

スをインプロセスで起動する場合の、J2EE サーバの起動手順について説明します。

なお、インプロセス HTTP サーバを起動する場合、および CORBA ネーミングサービスをインプロセスで起動する場合の前提を次に示します。

- インプロセス HTTP サーバを使用する場合
usrconf.properties ファイルの webserver.connector.inprocess_http.enabled キーに「true」が指定されている必要があります。
- J2EE サーバ用の CORBA ネーミングサービスをインプロセスで使用する場合
usrconf.properties ファイルの ejbserver.naming.port キーに J2EE サーバ用の CORBA ネーミングサービスの TCP ポート番号が設定されている必要があります。
ejbserver.naming.port = <TCP ポート番号 >

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%server%bin"
```

2. cjstartsv コマンドで J2EE サーバを開始します。
コマンドの実行には、Administrator 権限が必要です。

```
cjstartsv <サーバ名称>
```

(d) 標準システム構成で CORBA ネーミングサービスをアウトプロセスで起動する場合
標準システム構成で、J2EE サーバ用の CORBA ネーミングサービスをアウトプロセスで起動する場合の、J2EE サーバの起動手順について説明します。

なお、J2EE サーバ用の CORBA ネーミングサービスをアウトプロセスで起動する場合の前提を次に示します。

- J2EE サーバ起動前に、J2EE サーバ用の CORBA ネーミングサービスを起動しておく必要があります。CORBA ネーミングサービスの手動起動の方法については、「付録 D.2(5) CORBA ネーミングサービスの起動」を参照してください。
- usrconf.properties ファイルの ejbserver.naming.startupMode キーに「manual」を設定しておく必要があります。
- usrconf.properties ファイルの ejbserver.naming.port キーに設定する TCP ポート番号が、同じホストで起動しているほかの CORBA ネーミングサービスと同じ値にならないように注意してください。

J2EE サーバの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%server%bin"
```

2. cjstartsv コマンドで J2EE サーバを開始します。
コマンドの実行には、Administrator 権限が必要です。

```
cjstartsv <サーバ名称>
```

(9) リソースの開始

リソースの開始方法について説明します。

リソースは、サーバ管理コマンド (cjstarttrar コマンド, cjstartjb コマンド) を使用して開始します。

なお、開始するリソースは、あらかじめ J2EE リソースとして、J2EE サーバにデプロイされている必要があります。J2EE サーバへのリソースのデプロイも、サーバ管理コマンドで実行します。

J2EE アプリケーションに含まれるリソースアダプタを利用する場合、この操作は不要です。

詳細については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」のリソースの設定に関する説明、およびマニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」の J2EE リソースの設定に関する説明を参照してください。

! 注意事項

- Cosminexus RM を起動する場合、あらかじめ DB Connector for Cosminexus RM を起動しておいてください。DB Connector for Cosminexus RM を起動しないで Cosminexus RM を起動しようとするとエラーが発生します。
- ルートリソースアダプタは、ルートリソースアダプタに所属するすべてのメンバリソースアダプタが開始状態の場合に開始できます。

(10) J2EE アプリケーションの開始

J2EE アプリケーションの開始方法について説明します。

J2EE アプリケーションは、サーバ管理コマンドを使用して開始します。なお、サーバ管理コマンドの実行には、Administrator 権限が必要です。

J2EE アプリケーションは用途に応じて、動作モードを選択して開始します。本番稼働で動作する J2EE アプリケーションは通常モードで実行します。本番稼働前の J2EE アプリケーションの動作確認などで、テスト用として動作する J2EE アプリケーションはテストモードで実行します。

動作モードによって、J2EE アプリケーションの開始方法が異なります。それぞれの場合の開始方法を次に示します。

通常モードの場合

通常モードで J2EE アプリケーションを開始する場合、サーバ管理コマンド (cjstartapp) を使用します。

テストモードの場合

テストモードで J2EE アプリケーションを開始する場合、サーバ管理コマンドの cjstartapp コマンドに -test オプションを指定して実行します。

開始する J2EE アプリケーションは、J2EE サーバにインポートされている必要があります。J2EE アプリケーションとは、Enterprise Bean (EJB-JAR) およびサーブレットと JSP (WAR) を一つにまとめた EAR です。サーバ管理コマンドでは、J2EE アプリケーション単位に開始、停止を実行します。

J2EE サーバへの J2EE アプリケーションのインポートも、サーバ管理コマンドで実行します。また、開発環境で作成した WAR や EJB-JAR を、サーバ管理コマンドを使用して J2EE アプリケーションにすることもできます。

詳細については、マニュアル「Cosminexus アプリケーションサーバ システム構築・運用ガイド」の J2EE アプリケーションの開始に関する説明、およびマニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」の J2EE アプリケーションのインポートに関する説明を参照してください。

(11) Web サーバの起動

Web サーバの起動方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの起動のタイミングはシステムの運用形態に依存しますが、この順序で起動することで J2EE アプリケーションへのリクエストの受け付けを制御できます。Web サーバ起動後、Web クライアントからの J2EE アプリケーションへのリクエストの受け付けを開始します。

また、インプロセス HTTP サーバを使用する場合は、Web サーバの起動は必要ありません。

付録 D.3 システムの停止手順

ここでは、構成ソフトウェアの各プロセス、リソース、および J2EE アプリケーションの停止の手順について説明します。

(1) システムの停止の流れ

システムの停止手順は、起動手順と逆の順序になります。データベース、LDAP ディレクトリサーバ、EIS などを使用している場合は、必要に応じて最後に停止してください。

! 注意事項

システム構成によって、アプリケーションサーバだけを停止する場合と、アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを停止する場合があります。

アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを停止する場合、次の点に注意してください。

- セッションフェイルオーバーサーバを使用しているときは、アプリケーションサーバを停止したあとに、セッションフェイルオーバーサーバを停止してください。
- 統合ネーミングスケジューラサーバの停止順序は任意です。

次に示す停止操作を基にシステムを停止してください。

Web サーバの停止

停止方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの停止のタイミングについてはシステムの運用形態に依存しますが、最初に停止することで J2EE アプリケーションへのリクエストの受け付けを制御できます。また、インプロセス HTTP サーバを使用する場合は、Web サーバを停止する必要はありません。

J2EE アプリケーションの停止

サーバ管理コマンド (cjstopapp) で J2EE アプリケーションを停止します。停止方法については、「付録 D.4(2) J2EE アプリケーションの停止」を参照してください。

なお、次回 J2EE サーバの起動と同時に J2EE アプリケーションを開始する場合は、J2EE アプリケーションの停止は不要です。

リソースの停止

J2EE リソースを使用しない場合、この操作は不要です。

サーバ管理コマンド (cjstoprar, cjstopjb) で J2EE リソースを停止します。停止方法については、「付録 D.4(3) リソースの停止」を参照してください。

なお、次回 J2EE サーバ起動時に J2EE リソースを開始する場合は、J2EE リソースの停止は不要です。

J2EE サーバの停止

cjstopsv コマンドで J2EE サーバを停止します。停止方法については、「付録 D.4(4) J2EE サーバの停止」を参照してください。

SFO サーバの停止

cjstopsv コマンドで SFO サーバを停止します。SFO サーバは、J2EE サーバと同様の手順で停止します。停止方法については、「付録 D.4(5) SFO サーバの停止」を参照してください。

CTM デーモンの停止

CTM を使用しない場合、この操作は不要です。

ctmstop コマンドで CTM デーモンを停止します。停止方法については、「付録 D.4(6) CTM デーモンの停止」を参照してください。

CORBA ネーミングサービスの停止

CORBA ネーミングサービスの起動方法によって、停止方法が異なります。停止方法については、「付録 D.4(7) CORBA ネーミングサービスの停止」を参照してください。

CTM ドメインマネージャの停止

CTM を使用しない場合、この操作は不要です。

ctmdmstop コマンドで CTM ドメインマネージャを停止します。停止方法については、「付録 D.4(8) CTM ドメインマネージャの停止」を参照してください。

スマートエージェントの停止

スマートエージェントを起動していない場合、この操作は不要です。

停止方法については、「付録 D.4(9) スマートエージェントの停止」を参照してください。

PRF デーモンの停止

cpfstop コマンドで PRF デーモンを停止します。停止方法については、「付録 D.4(10) PRF デーモンの停止」を参照してください。

付録 D.4 システムの停止方法

ここでは、構成ソフトウェアの各プロセスの停止方法、リソースの停止方法、および J2EE アプリケーションの停止方法について説明します。

(1) Web サーバの停止

Web サーバの停止方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの停止のタイミングについてはシステムの運用形態に依存しますが、最初に停止することで J2EE アプリケーションへのリクエストの受け付けを制御できます。また、インプロセス HTTP サーバを使用する場合は、Web サーバを停止する必要はありません。

(2) J2EE アプリケーションの停止

J2EE アプリケーションの停止方法について説明します。

J2EE アプリケーションは、サーバ管理コマンドを使用して停止します。通常モードの場合、およびテストモードの場合の停止方法を次に示します。

通常モードの場合

通常モードの J2EE アプリケーションを停止する場合、サーバ管理コマンド (cjstopapp) を使用します。

テストモードの場合

テストモードの J2EE アプリケーションを停止する場合、CUI を使用します。

cjstopapp コマンドに `-test` オプションを指定して実行します。

なお、サーバ管理コマンドの実行には、Administrator 権限が必要です。

J2EE アプリケーションを停止しても、J2EE アプリケーション内のリクエスト処理が終了しないなどの原因で停止処理が完了しない場合があります。この場合、J2EE アプリケーションを強制的に停止する必要があります。J2EE アプリケーションを強制停止する場合、cjstopapp コマンドに `-force` オプションを指定して実行します。J2EE アプリケーションを停止するための具体的な手順については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」を参照してください。

(3) リソースの停止

リソースの停止方法について説明します。

リソースの停止は、サーバ管理コマンド (cjstoprar, cjstopjb) を使用します。

なお、サーバ管理コマンドの実行には、Administrator 権限が必要です。

J2EE アプリケーションに含まれるリソースアダプタを利用する場合、この操作は不要です。

! 注意事項

- DB Connector for Cosminexus RM を停止する場合、あらかじめ Cosminexus RM を停止しておいてください。Cosminexus RM を停止しないで DB Connector for Cosminexus RM を停止しようとするとエラーが発生します。
- メンバリソースアダプタは、所属するルートリソースアダプタが停止状態のときだけ停止できます。
- 別名を定義した J2EE リソースは、J2EE サーバで開始されている J2EE アプリケーションがある場合は停止できません。停止しようとするとメッセージが出力されて処理が失敗します。別名を定義した J2EE リソースは、J2EE サーバで開始されているすべての J2EE アプリケーションを停止してから停止してください。

(4) J2EE サーバの停止

J2EE サーバの停止の手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%server%bin"
```

2. cjstopsv コマンドで J2EE サーバを停止します。
コマンドの実行には、Administrator 権限が必要です。

```
cjstopsv <サーバ名称>
```

cjstopsv コマンドの実行例を次に示します。この例では、Cosminexus のインストールディレクトリは「C:\Program Files\Hitachi\Cosminexus」、サーバ名は「MyServer」です。

```
C:>cd "C:\Program Files\HITACHI\Cosminexus\CC\server\bin"
C:\Program Files\HITACHI\Cosminexus\CC\server\bin>cjstopsv MyServer
C:\Program Files\HITACHI\Cosminexus\CC\server\bin>
```

なお、J2EE サーバを停止しても、J2EE サーバでの処理が終了しないなどの原因で停止処理が完了しない場合があります。この場合、J2EE サーバを強制的に停止する必要があります。J2EE サーバを強制停止する場合、cjstopsv コマンドに -f オプションまたは -fd オプションを指定して実行します。-fd オプションを指定すると、J2EE サーバの障害情報を取得したあと、J2EE サーバを強制停止します。

注意事項

CORBA ネーミングサービスを自動起動している場合は、J2EE サーバ停止時に CORBA ネーミングサービスも自動停止されます。

(5) SFO サーバの停止

SFO サーバは、J2EE サーバと同様の手順で停止します。

J2EE サーバの停止手順については、「付録 D.4(4) J2EE サーバの停止」を参照してください。

(6) CTM デーモンの停止

CTM デーモンの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>\CTM\bin"
```

2. ctmstop コマンドで CTM デーモンを停止します。
コマンドの実行には、Administrator 権限が必要です。

```
ctmstop
```

なお、CTM デーモンを停止しても、CTM デーモンの停止処理が完了しない場合があります。この場合、CTM デーモンを強制的に停止する必要があります。CTM デーモンを強制停止する場合、ctmstop コマンドに -CTMForce オプションを指定して実行します。

(7) CORBA ネーミングサービスの停止

CORBA ネーミングサービスの停止手順は、CORBA ネーミングサービスの起動方法によって異なります。

- J2EE サーバ起動時にインプロセスで起動している場合
J2EE サーバ停止時に CORBA ネーミングサービスも自動停止します。
- アウトプロセスで起動している場合
 1. J2EE サーバを停止しておきます。
J2EE サーバの停止については、「付録 D.4(4) J2EE サーバの停止」を参照してください。
 2. [Ctrl] + [C] キーで、CORBA ネーミングサービスを手動停止します。

(8) CTM ドメインマネージャの停止

CTM ドメインマネージャの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CTM%bin"
```

2. ctmdmstop コマンドで CTM ドメインマネージャを停止します。
コマンドの実行には、Administrator 権限が必要です。

```
ctmdmstop
```

なお、CTM ドメインマネージャを停止しても、CTM ドメインマネージャの停止処理が完了しない場合があります。この場合、CTM ドメインマネージャを強制的に停止する必要があります。CTM ドメインマネージャを強制停止する場合、ctmdmstop コマンドに -CTMForce オプションを指定して実行します。

(9) スマートエージェントの停止

スマートエージェントの停止手順を次に示します。

1. Windows のタスクバーに表示されているスマートエージェントのアイコンを右クリックして [Exit] を選択し、スマートエージェントを停止します。

(10) PRF デーモンの停止

PRF デーモンの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%PRF%bin"
```

2. cprfstop コマンドで PRF デーモンを停止します。
cprfstart コマンドを実行したユーザが実行してください

```
cprfstop
```

なお、PRF デーモンを停止しても、PRF デーモンの停止処理が完了しない場合があります。この場合、PRF デーモンを強制的に停止する必要があります。PRF デーモンを強制停止する場合、cprfstop コマンドに `-Force` オプションを指定して実行します。

付録 E J2EE アプリケーションを実行するシステムの起動と停止 (UNIX の場合)

ここでは、UNIX の場合の J2EE アプリケーションを実行するシステムの起動と停止の手順について説明します。

付録 E.1 システムの起動手順

システムの起動手順について説明します。

! 注意事項

システムでは、システム構成によって、アプリケーションサーバだけを起動する場合と、アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを起動する場合があります。

アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを起動する場合、次の点に注意してください。

- セッションフェイルオーバーサーバを使用するときは、アプリケーションサーバより先にセッションフェイルオーバーサーバを起動してください。
- 統合ネーミングスケジューラサーバの起動順序は任意です。

以降の項では、アプリケーションサーバ、統合ネーミングスケジューラサーバおよびセッションフェイルオーバーサーバの起動手順について説明します。

なお、データベース、LDAP ディレクトリサーバ、EIS などの関連製品は起動済みであることを前提に説明します。データベース、LDAP ディレクトリサーバ、EIS などを使用している場合は、必要に応じてあらかじめ起動しておいてください。

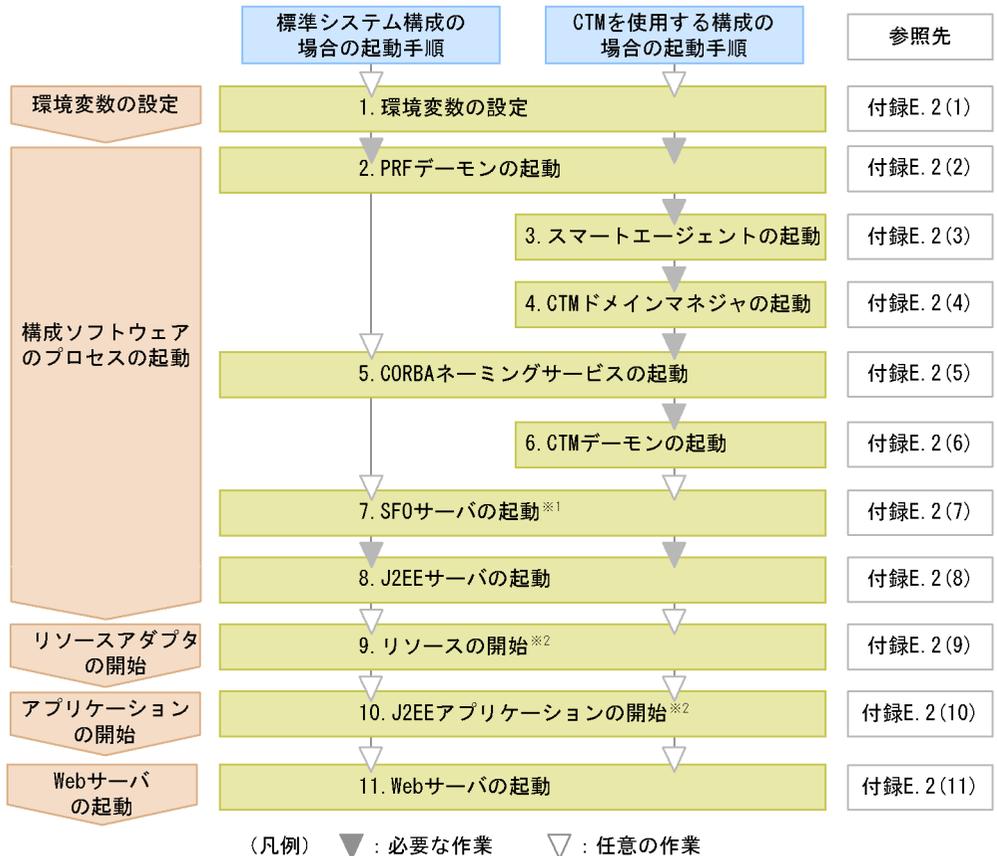
(1) アプリケーションサーバの起動手順の流れ

システムで使用するアプリケーションサーバの起動手順の流れについて説明します。

ここでは、標準システム構成の場合と、CTM を使用する構成の場合のアプリケーションサーバの起動手順の流れについて説明します。

アプリケーションサーバの起動手順の流れを次の図に示します。

図 E-1 アプリケーションサーバの起動の流れ (UNIX の場合)



注※1 SF0サーバを起動したときに、SF0サーバアプリケーションが停止している場合、SF0サーバアプリケーションを開始します。

注※2 J2EEサーバを起動したときに、リソースまたはJ2EEアプリケーションが停止している場合だけ必要な手順です。

図中の 1. ~ 11. について説明します。

1. 環境変数の設定

環境変数を設定します。環境変数の設定方法については、「付録 E.2(1) システム起動時の環境変数の設定」を参照してください。

2. PRF デーモンの起動

cpfststart コマンドで PRF デーモンを起動します。起動方法については、「付録 E.2(2) PRF デーモンの起動」を参照してください。

3. スマートエージェントの起動

CTM を使用しない場合、この操作は不要です。

osagent コマンドでスマートエージェントを起動します。起動方法については、「付録

E.2(3) スマートエージェントの起動」を参照してください。

4. CTM ドメインマネージャの起動

CTM を使用しない場合、この操作は不要です。

ctmdmstart コマンドで CTM ドメインマネージャを起動します。起動方法については、「付録 E.2(4) CTM ドメインマネージャの起動」を参照してください。

5. CORBA ネーミングサービスの起動

J2EE サーバ用の CORBA ネーミングサービスは、J2EE サーバ起動時にインプロセスで起動することを推奨します。

アウトプロセスでの起動方法については、「付録 E.2(5) CORBA ネーミングサービスの起動」を参照してください。

6. CTM デーモンの起動

CTM を使用しない場合、この操作は不要です。

ctmstart コマンドで CTM デーモンを起動します。起動方法については、「付録 E.2(6) CTM デーモンの起動」を参照してください。

7. SFO サーバの起動

SFO サーバを起動しない場合、この操作は不要です。また、アプリケーションサーバ内で SFO サーバを起動しない場合もこの操作は不要です。

cjstartsv コマンドに `-nosecurity` オプションを指定して、SFO サーバを起動します。また、SFO サーバを起動したときに SFO サーバアプリケーションが停止している場合、サーバ管理コマンド (`cjstartapp`) で SFO サーバアプリケーションを開始します。

起動方法およびアプリケーションの開始については、「付録 E.2(7) SFO サーバの起動」を参照してください。

8. J2EE サーバの起動

cjstartsv コマンドで J2EE サーバを起動します。起動方法については、「付録 E.2(8) J2EE サーバの起動」を参照してください。

9. リソースの開始

リソースを使用しない場合、この操作は不要です。

また、J2EE アプリケーションに含まれるリソースアダプタを利用する場合もこの操作は不要です。

サーバ管理コマンドでリソースを開始します。開始方法については、「付録 E.2(9) リソースの開始」を参照してください。

10. J2EE アプリケーションの開始

サーバ管理コマンドで J2EE アプリケーションを開始します。なお、リソースアダプタを含んだ J2EE アプリケーションの場合、J2EE アプリケーションに含まれるすべてのリソースアダプタが同時に開始されます。開始方法については、「付録 E.2(10) J2EE アプリケーションの開始」を参照してください。

11. Web サーバの起動

起動方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの起動のタイミングはシステムの運用形態に依存しますが、この順序で起動することで J2EE アプリケーションへのリクエストの受け付けを制御できます。Web サーバ起動後、Web クライアントからの J2EE アプリケーションへのリクエストの受け付けを開始します。開始方法については、「付録 E.2(11) Web サーバの起動」を参照してください。

また、インプロセス HTTP サーバを使用する場合は、Web サーバの起動は必要ありません。

(2) 統合ネーミングスケジューラサーバの起動の流れ

統合ネーミングスケジューラサーバの起動の流れを次の図に示します。

図 E-2 統合ネーミングスケジューラサーバの起動の流れ (UNIX の場合)



図中の 1. ~ 6. について説明します。

1. 環境変数の設定

環境変数を設定します。環境変数の設定方法については、「付録 E.2(1) システム起動時の環境変数の設定」を参照してください。

2. PRF デーモンの起動

cprfstart コマンドで PRF デーモンを起動します。起動方法については、「付録 E.2(2) PRF デーモンの起動」を参照してください。

3. スマートエージェントの起動

osagent コマンドでスマートエージェントを起動します。起動方法については、「付録 E.2(3) スマートエージェントの起動」を参照してください。

4. CTM ドメインマネージャの起動

ctmdmstart コマンドで CTM ドメインマネージャを起動します。起動方法については、「付録 E.2(4) CTM ドメインマネージャの起動」を参照してください。

5. CORBA ネーミングサービスの起動

CTM 用のグローバル CORBA ネーミングサービスを起動します。起動方法については、「付録 E.2(5) CORBA ネーミングサービスの起動」を参照してください。

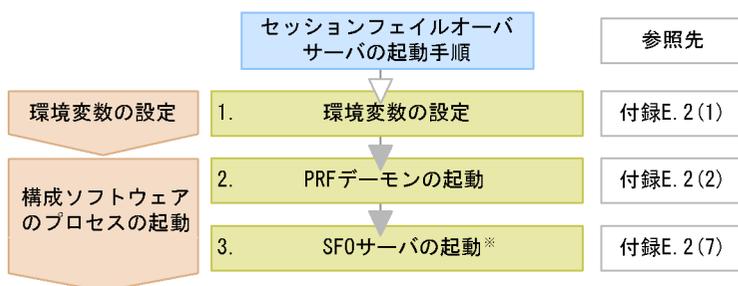
6. CTM デーモンの起動

ctmstart コマンドで CTM デーモンを起動します。起動方法については、「付録 E.2(6) CTM デーモンの起動」を参照してください。

(3) セッションフェイルオーバーサーバの起動の流れ

セッションフェイルオーバーサーバの起動の流れを次の図に示します。

図 E-3 セッションフェイルオーバーサーバの起動の流れ (UNIX の場合)



注※ SF0サーバを起動したときに、SF0サーバアプリケーションが停止している場合、SF0サーバアプリケーションを開始します。

図中の 1. ~ 3. について説明します。

1. 環境変数の設定

環境変数を設定します。環境変数の設定方法については、「付録 E.2(1) システム起動時の環境変数の設定」を参照してください。

2. PRF デーモンの起動

cprfststart コマンドで PRF デーモンを起動します。起動方法については、「付録 E.2(2) PRF デーモンの起動」を参照してください。

3. SFO サーバの起動

cjstartsv コマンドに `-nosecurity` オプションを指定して、SFO サーバを起動します。また、SFO サーバを起動したときに、SFO サーバアプリケーションが停止している場合、サーバ管理コマンド (`cjstartapp`) で SFO サーバアプリケーションを開始します。

なお、SFO サーバでは、明示管理ヒープ機能の設定を無効にすることを推奨します。

起動方法およびアプリケーションの開始については、「付録 E.2(7) SFO サーバの起動」を参照してください。

！ 注意事項

SFO サーバで実行できるアプリケーションは、SFO サーバアプリケーションだけです。SFO サーバで SFO サーバアプリケーション以外のアプリケーションは実行しないでください。

付録 E.2 システムの起動方法

ここでは、構成ソフトウェアの各プロセスの起動方法、リソースの開始方法、および J2EE アプリケーションの開始方法について説明します。

(1) システム起動時の環境変数の設定

ここでは、アプリケーションサーバ、統合ネーミングスケジューラサーバ、および SFO サーバに設定する環境変数について説明します。

C シェルの場合、setenv コマンドで次に示す環境変数を設定してください。

```
set path=( /opt/Cosminexus/jdk/bin /opt/Cosminexus/TPB/bin /bin $path)
setenv SHLIB_PATH "${SHLIB_PATH}:/opt/Cosminexus/TPB/lib:/opt/Cosminexus/PRF/lib:/opt/hitachi/common/lib"
setenv VBROKER_ADM /opt/Cosminexus/TPB/adm
setenv TPDIR /opt/Cosminexus/TPB
setenv PRFSPPOOL /opt/Cosminexus/PRF/spool
setenv CTMDIR /opt/Cosminexus/CTM
setenv CTMSPPOOL /opt/Cosminexus/CTM/spool
setenv TZ JST-9
```

注

CTM を使用する場合に、アプリケーションサーバおよび統合ネーミングスケジューラサーバで設定が必要な環境変数です。

- 下線の部分は、OS ごとに変数名が異なります。AIX の場合は「LIBPATH」、HP-UX (IPF)、Linux および Solaris の場合は「LD_LIBRARY_PATH」となります。
- CORBA ネーミングサービスを手動起動する場合は、環境変数「PATH」の先頭に「/opt/Cosminexus/jdk/bin」を指定してください。
- HiRDB または Oracle を利用する場合には、それぞれのライブラリのパスを環境変数「LIBPATH」(AIX の場合)、または「LD_LIBRARY_PATH」(HP-UX、Linux および Solaris の場合) に設定します。あらかじめ環境変数に設定している場合は、setenv コマンドの実行は不要です。

(2) PRF デーモンの起動

PRF デーモンの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/PRF/bin
```

2. cprfstart コマンドで PRF デーモンを起動します。
Solaris の場合、コマンドの実行には、root 権限が必要です。

```
# cprfstart
```

(3) スマートエージェントの起動

スマートエージェントの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/TPB/bin
```

2. osagent コマンドでスマートエージェントを起動します。

```
# osagent &
```

(4) CTM ドメインマネージャの起動

CTM ドメインマネージャの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CTM/bin
```

2. ctmdmstart コマンドで CTM ドメインマネージャを起動します。
Solaris の場合、コマンドの実行には、root 権限が必要です。

```
# ctmdmstart
```

(5) CORBA ネーミングサービスの起動

CORBA ネーミングサービスの起動方法について説明します。

J2EE サーバ用の CORBA ネーミングサービスは、J2EE サーバ起動時にインプロセスで起動することを推奨します。アウトプロセスで起動する場合は、nameserv コマンドを使用して起動します。

ここでは、nameserv コマンドで CORBA ネーミングサービスをアウトプロセスで起動

する方法について説明します。J2EE サーバの起動時に CORBA ネーミングサービスをインプロセスで起動する方法については、「付録 E.2(8) J2EE サーバの起動」を参照してください。

なお、CORBA ネーミングサービス起動時は次の点に留意してください。

- CTM を使用する場合、J2EE サーバ用および CTM 用の CORBA ネーミングサービスを起動します。CTM 用にグローバル CORBA ネーミングサービスを nameserv コマンドで手動起動して、さらに J2EE サーバ用の CORBA ネーミングサービスを J2EE サーバ起動時にインプロセスで自動起動してください。
- J2EE サーバ間でのセッション情報の引き継ぎを行う場合、アプリケーションサーバ内で SFO サーバを起動するときは、SFO サーバ用の CORBA ネーミングサービスはインプロセスで起動することが前提のため起動は不要です。

nameserv コマンドで CORBA ネーミングサービスを手動起動する場合の手順を次に示します。

1. unsetenv コマンドで環境変数「CLASSPATH」を設定解除します。(C シェルの場合)

```
# unsetenv CLASSPATH
```

2. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/TPB/bin
```

3. nameserv コマンドでローカル CORBA ネーミングサービスを手動開始します。コマンドの実行には、root 権限が必要です。

```
# nameserv -J-Dvbroker.agent.enableLocator=false
-J-Djava.security.policy==/opt/Cosminexus/CC/server/sysconf/cli.policy
-J-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<ポート番号> &
```

注意事項

- CORBA ネーミングサービスを手動起動する場合は、環境変数「PATH」の先頭に「/opt/Cosminexus/jdk/bin」を指定してください。
- nameserv コマンドで指定するオプションの値に空白を含むディレクトリが含まれる場合は、値全体をダブルクォーテーション (") で囲むようにしてください。
- nameserv コマンドで指定する値は環境に合わせて変更してください。
- CORBA ネーミングサービスは、次のオプションで指定したポート番号で起動します。
-J-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=<ポート番号>

(6) CTM デーモンの起動

CTM デーモンの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CTM/bin
```

2. ctmstart コマンドで CTM デーモンを起動します。

Solaris の場合、コマンドの実行には、root 権限が必要です。

```
# ctmstart -CTMINSRef <ホスト名:ポート番号>
```

ホスト名およびポート番号にはグローバル CORBA ネーミングサービスを起動しているホスト名およびポート番号を指定してください。

参考

CTM デーモンを開始する ctmstart コマンドでは、コマンド実行時に次のファイルを指定できます。

- CTM ユーザ環境変数定義ファイル
-CTMEnviron オプションで指定するファイルです。CTM デーモンで管理するプロセスが使用する環境変数を指定したい場合は、このファイルに環境変数を記述します。
- CTM コマンドオプションファイル
-CTMRegOption で指定するファイルです。CTM レギュレータを自動開始するときのコマンドオプションを指定したい場合は、このファイルにコマンドオプションを記述します。

必要に応じて、これらのファイルを作成しておいてください。

(7) SFO サーバの起動

SFO サーバの起動手順を次に示します。

! 注意事項

- cjstartsv コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。
- root 権限を持たないユーザを Component Container 管理者に設定している場合、SFO サーバ用の CORBA ネーミングサービスのポート番号に 1024 以上の値を指定しておく必要があります。1024 以上の値が指定されていない場合、エラーとなり、SFO サーバは起動できません。
- SFO サーバ起動時に KDJE47500-E というエラーメッセージが出力された場合、利用しようとした CORBA ネーミングサービスがすでに起動されているため、SFO サーバの起動処理に失敗します。すでに起動している CORBA ネーミングサービスを停止して、SFO サーバを起動してください。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. cjstartsv コマンドに -nosecurity オプションを指定して、SFO サーバを開始します。コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
# cjstartsv <サーバ名称> -nosecurity
```

3. SFO サーバアプリケーションを開始します。
SFO サーバを起動したときに SFO サーバアプリケーションが停止している場合、サーバ管理コマンドを使用して、SFO サーバアプリケーションを開始します。なお、サーバ管理コマンドの実行には、root 権限、または Component Container 管理者の権限が必要です。
SFO サーバアプリケーションの名称は「CCC_SFO_Server_Application」です。

! 注意事項

SFO サーバで実行できるアプリケーションは、SFO サーバアプリケーションだけです。SFO サーバで SFO サーバアプリケーション以外のアプリケーションは実行しないでください。

(8) J2EE サーバの起動

J2EE サーバの起動方法について説明します。

J2EE サーバ起動時に、次に示すプロセスをインプロセスで起動することを推奨します。

- J2EE サーバ用の CORBA ネーミングサービス
- トランザクションサービス (トランザクションサービスを使用する場合)

なお, CTM を使用する場合, J2EE サーバ用の CORBA ネーミングサービスはインプロセスで起動してください。

次に示す場合の J2EE サーバの起動手順について (a) ~ (d) で説明します。

- 標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合
- CTM を使用する構成で CORBA ネーミングサービスおよびトランザクションサービスをインプロセスで起動する場合
- 標準システム構成でインプロセス HTTP サーバを起動して CORBA ネーミングサービスをインプロセスで起動する場合
- 標準システム構成で CORBA ネーミングサービスをアウトプロセスで起動する場合

また, J2EE サーバ起動時は次の点に留意してください。

- 複数の J2EE サーバで一つのネーミングサービスを共有する場合, J2EE サーバは一つずつ起動してください。複数の J2EE サーバを同時に起動しようとする J2EE サーバの起動に失敗することがあります。
- `cjstartsv` コマンドの実行には, root 権限 (Component Container 管理者を設定していない場合), または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。
- root 権限を持たないユーザを Component Container 管理者に設定している場合, CORBA ネーミングサービスを起動するときは, J2EE サーバ用の CORBA ネーミングサービスのポート番号に 1024 以上の値を指定しておく必要があります。1024 以上の値が指定されていない場合, エラーとなり, J2EE サーバは起動できません。
- J2EE サーバ起動時に KDJE47500-E というエラーメッセージが出力された場合, 利用しようとした CORBA ネーミングサービスがすでに起動されているため, J2EE サーバの起動処理に失敗します。すでに起動している CORBA ネーミングサービスを停止して, J2EE サーバを起動してください。

(a) 標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合

標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合の, J2EE サーバの起動手順を次に示します。

1. `cd` コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. `cjstartsv` コマンドで J2EE サーバを開始します。

コマンドの実行には, root 権限 (Component Container 管理者を設定していない場合), または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
# cjstartsv <サーバ名称>
```

(b) CTM を使用する構成で CORBA ネーミングサービスおよびトランザクションサービスをインプロセスで起動する場合

CTM を使用する構成で、J2EE サーバ用の CORBA ネーミングサービス、およびトランザクションサービスをインプロセスで起動する場合の、J2EE サーバの起動手順について説明します。

なお、J2EE サーバ用の CORBA ネーミングサービス、およびトランザクションサービスをインプロセスで起動する場合の前提を次に示します。

- usrconf.properties ファイルの ejbserver.naming.port キーに J2EE サーバ用の CORBA ネーミングサービスの TCP ポート番号が設定されている必要があります。
ejbserver.naming.port = <TCP ポート番号 >

J2EE サーバの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. cjstartsv コマンドで J2EE サーバを開始します。

コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
# cjstartsv <サーバ名称>
```

(c) 標準システム構成でインプロセス HTTP サーバを起動して CORBA ネーミングサービスをインプロセスで起動する場合

標準システム構成でインプロセス HTTP サーバを起動して、CORBA ネーミングサービスをインプロセスで起動する場合の、J2EE サーバの起動手順について説明します。

なお、インプロセス HTTP サーバを起動する場合、および CORBA ネーミングサービスをインプロセスで起動する場合の前提を次に示します。

- インプロセス HTTP サーバを使用する場合
usrconf.properties ファイルの webserver.connector.inprocess_http.enabled キーに「true」が指定されている必要があります。
- J2EE サーバ用の CORBA ネーミングサービスをインプロセスで使用する場合
usrconf.properties ファイルの ejbserver.naming.port キーに J2EE サーバ用の

CORBA ネーミングサービスの TCP ポート番号が設定されている必要があります。

`ejbserver.naming.port = <TCP ポート番号 >`

1. `cd` コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. `cjstartsv` コマンドで J2EE サーバを開始します。

コマンドの実行には、`root` 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
# cjstartsv <サーバ名称>
```

(d) 標準システム構成で CORBA ネーミングサービスをアウトプロセスで起動する場合
標準システム構成で、J2EE サーバ用の CORBA ネーミングサービスをアウトプロセスで起動する場合の、J2EE サーバの起動手順について説明します。

なお、J2EE サーバ用の CORBA ネーミングサービスをアウトプロセスで起動する場合の前提を次に示します。

- J2EE サーバ起動前に、J2EE サーバ用の CORBA ネーミングサービスを起動しておく必要があります。CORBA ネーミングサービスの手動起動の方法については、「付録 E.2(5) CORBA ネーミングサービスの起動」を参照してください。
- `usrconf.properties` ファイルの `ejbserver.naming.startupMode` キーに「`manual`」を設定しておく必要があります。
- `usrconf.properties` ファイルの `ejbserver.naming.port` キーに設定する TCP ポート番号が、同じホストで起動しているほかの CORBA ネーミングサービスと同じ値にならないように注意してください。

J2EE サーバの起動手順を次に示します。

1. `cd` コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. `cjstartsv` コマンドで J2EE サーバを開始します。

コマンドの実行には、`root` 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
# cjstartsv <サーバ名称>
```

(9) リソースの開始

リソースの開始方法について説明します。

リソースは、サーバ管理コマンド (cjstarttrrar コマンド, cjstartjb コマンド) を使用して開始します。

なお、開始するリソースは、あらかじめ J2EE リソースとして、J2EE サーバにデプロイされている必要があります。J2EE サーバへのリソースのデプロイも、サーバ管理コマンドで実行します。

J2EE アプリケーションに含まれるリソースアダプタを利用する場合、この操作は不要です。

詳細については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」のリソースの設定に関する説明、およびマニュアル「Cosminexus アプリケーションサーバアプリケーション設定操作ガイド」の J2EE リソースの設定に関する説明を参照してください。

! 注意事項

- Cosminexus RM を起動する場合、あらかじめ DB Connector for Cosminexus RM を起動しておいてください。DB Connector for Cosminexus RM を起動しないで Cosminexus RM を起動しようとするとエラーが発生します。
- ルートリソースアダプタは、ルートリソースアダプタに所属するすべてのメンバリソースアダプタが開始状態の場合に開始できます。

(10) J2EE アプリケーションの開始

J2EE アプリケーションの開始方法について説明します。

J2EE アプリケーションは、サーバ管理コマンドを使用して開始します。なお、サーバ管理コマンドの実行には、root 権限、または Component Container 管理者の権限が必要です。

J2EE アプリケーションは用途に応じて、動作モードを選択して開始します。本番稼働で動作する J2EE アプリケーションは通常モードで実行します。本番稼働前の J2EE アプリケーションの動作確認などで、テスト用として動作する J2EE アプリケーションはテストモードで実行します。

動作モードによって、J2EE アプリケーションの開始方法が異なります。それぞれの場合の開始方法を次に示します。

通常モードの場合

通常モードで J2EE アプリケーションを開始する場合、サーバ管理コマンドの cjstartapp コマンドを使用します。

テストモードの場合

テストモードで J2EE アプリケーションを開始する場合、サーバ管理コマンドの `cjstartapp` コマンドに `-test` オプションを指定して実行します。

開始する J2EE アプリケーションは、あらかじめ J2EE アプリケーションとして、J2EE サーバにインポートされている必要があります。J2EE アプリケーションとは、アプリケーション開発環境で作成した J2EE アプリケーションの構成要素である Enterprise Bean (EJB-JAR) およびサーブレットと JSP (WAR) を一つにまとめた EAR です。サーバ管理コマンドでは、J2EE アプリケーション単位に開始、停止を実行します。

J2EE サーバへの J2EE アプリケーションのインポートも、サーバ管理コマンドで実行します。また、開発環境で作成した WAR や EJB-JAR を、サーバ管理コマンドを使用して J2EE アプリケーションにすることもできます。

詳細については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」の J2EE アプリケーションの設定に関する説明、およびマニュアル「Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド」の J2EE アプリケーションのインポートに関する説明を参照してください。

(11) Web サーバの起動

Web サーバの起動方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの起動のタイミングはシステムの運用形態に依存しますが、この順序で起動することで J2EE アプリケーションへのリクエストの受け付けを制御できます。Web サーバ起動後、Web クライアントからの J2EE アプリケーションへのリクエストの受け付けを開始します。

また、インプロセス HTTP サーバを使用する場合は、Web サーバの起動は必要ありません。

付録 E.3 システムの停止手順

ここでは、構成ソフトウェアの各プロセス、リソース、および J2EE アプリケーションの停止の手順について説明します。

(1) システムの停止の流れ

システムの停止手順は、起動手順と逆の順序になります。データベース、LDAP ディレクトリサーバ、EIS などを使用している場合は、必要に応じて最後に停止してください。

! 注意事項

システム構成によって、アプリケーションサーバだけを停止する場合と、アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを停止する場合があります。

アプリケーションサーバのほかに、統合ネーミングスケジューラサーバ、セッションフェイルオーバーサーバを停止する場合、次の点に注意してください。

- セッションフェイルオーバーサーバを使用しているときは、アプリケーションサーバを停止したあとに、セッションフェイルオーバーサーバを停止してください。
- 統合ネーミングスケジューラサーバの停止順序は任意です。

次に示す停止操作を基にシステムを停止してください。

Web サーバの停止

停止方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの停止のタイミングについてはシステムの運用形態に依存しますが、最初に停止することで J2EE アプリケーションへのリクエストの受け付けを制御できます。また、インプロセス HTTP サーバを使用する場合は、Web サーバを停止する必要はありません。

J2EE アプリケーションの停止

サーバ管理コマンド (cjstopapp) で J2EE アプリケーションを停止します。停止方法については、「付録 E.4(2) J2EE アプリケーションの停止」を参照してください。

なお、次回 J2EE サーバの起動と同時に J2EE アプリケーションを開始する場合は、J2EE アプリケーションの停止は不要です。

リソースの停止

J2EE リソースを使用しない場合、この操作は不要です。

サーバ管理コマンド (cjstoprar) で J2EE リソースを停止します。停止方法については、「付録 E.4(3) リソースの停止」を参照してください。

なお、次回 J2EE サーバ起動時に J2EE リソースを開始する場合は、J2EE リソースの停止は不要です。

J2EE サーバの停止

cjstopsv コマンドで J2EE サーバを停止します。停止方法については、「付録 E.4(4) J2EE サーバの停止」を参照してください。

SFO サーバの停止

cjstopsv コマンドで SFO サーバを停止します。SFO サーバは、J2EE サーバと同様の手順で停止します。停止方法については、「付録 E.4(4) J2EE サーバの停止」を参照してください。

CTM デーモンの停止

CTM を使用しない場合、この操作は不要です。

ctmstop コマンドで CTM デーモンを停止します。停止方法については、「付録 E.4(6) CTM デーモンの停止」を参照してください。

CORBA ネーミングサービスの停止

CORBA ネーミングサービスの起動方法によって、停止方法が異なります。停止方法については、「付録 E.4(7) CORBA ネーミングサービスの停止」を参照してください。

CTM ドメインマネージャの停止

CTM を使用しない場合、この操作は不要です。

ctmdmstop コマンドで CTM ドメインマネージャを停止します。停止方法については、「付録 E.4(8) CTM ドメインマネージャの停止」を参照してください。

スマートエージェントの停止

スマートエージェントを起動していない場合、この操作は不要です。

停止方法については、「付録 E.4(9) スマートエージェントの停止」を参照してください。

PRF デーモンの停止

cprfstop コマンドで PRF デーモンを停止します。停止方法については、「付録 E.4(10) PRF デーモンの停止」を参照してください。

付録 E.4 システムの停止方法

ここでは、構成ソフトウェアの各プロセスの停止方法、リソースの停止方法、および J2EE アプリケーションの停止方法について説明します。

(1) Web サーバの停止

Web サーバの停止方法については、ご使用の Web サーバのマニュアルを参照してください。

なお、Web サーバの停止のタイミングについてはシステムの運用形態に依存しますが、最初に停止することで J2EE アプリケーションへのリクエストの受け付けを制御できます。また、インプロセス HTTP サーバを使用する場合は、Web サーバを停止する必要はありません。

(2) J2EE アプリケーションの停止

J2EE アプリケーションの停止方法について説明します。

J2EE アプリケーションは、サーバ管理コマンドを使用して停止します。通常モードの場合、およびテストモードの場合の停止方法を次に示します。

通常モードの場合

通常モードの J2EE アプリケーションを停止する場合、サーバ管理コマンド (cjstopapp) を使用します。

テストモードの場合

テストモードの J2EE アプリケーションを停止する場合、サーバ管理コマンド

(`cjstopapp`) に `-test` オプションを指定して実行します。

なお、サーバ管理コマンドの実行には、`root` 権限または Component Container 管理者の権限が必要です。

J2EE アプリケーションを停止しても、J2EE アプリケーション内のリクエスト処理が終了しないなどの原因で停止処理が完了しない場合があります。この場合、J2EE アプリケーションを強制的に停止する必要があります。J2EE アプリケーションを強制停止する場合、`cjstopapp` コマンドに `-force` オプションを指定して実行します。J2EE アプリケーションを停止するための具体的な手順については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」を参照してください。

(3) リソースの停止

リソースの停止方法について説明します。

リソースの停止は、サーバ管理コマンド (`cjstoprar`, `cjstopjb`) を使用します。

なお、サーバ管理コマンドの実行には、`root` 権限または Component Container 管理者の権限が必要です。

J2EE アプリケーションに含まれるリソースアダプタを利用する場合、この操作は不要です。

! 注意事項

- DB Connector for Cosminexus RM を停止する場合、あらかじめ Cosminexus RM を停止しておいてください。Cosminexus RM を停止しないで DB Connector for Cosminexus RM を停止しようとするとエラーが発生します。
- メンバリソースアダプタは、所属するルートリソースアダプタが停止状態のときだけ停止できます。
- 別名を定義した J2EE リソースは、J2EE サーバで開始されている J2EE アプリケーションがある場合は停止できません。停止しようとするとメッセージが出力されて処理が失敗します。別名を定義した J2EE リソースは、J2EE サーバで開始されているすべての J2EE アプリケーションを停止してから停止してください。

(4) J2EE サーバの停止

J2EE サーバの停止の手順を次に示します。

1. `cd` コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. `cjstopsv` コマンドで J2EE サーバを停止します。
コマンドの実行には、`root` 権限 (Component Container 管理者を設定していない場

合), または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
# cjstopsv <サーバ名称>
```

cjstopsv コマンドの実行例を次に示します。この例では、サーバ名は「MyServer」です。

```
# cd /opt/Cosminexus/CC/server/bin
# cjstopsv MyServer
#
```

なお、J2EE サーバを停止しても、J2EE サーバでの処理が終了しないなどの原因で停止処理が完了しない場合があります。この場合、J2EE サーバを強制的に停止する必要があります。J2EE サーバを強制停止する場合、cjstopsv コマンドに `-f` オプションまたは `-fd` オプションを指定して実行します。`-fd` オプションを指定すると、J2EE サーバの障害情報を取得したあと、J2EE サーバを強制停止します。

注意事項

CORBA ネーミングサービスを自動起動している場合は、J2EE サーバ停止時に CORBA ネーミングサービスも自動停止されます。

(5) SFO サーバの停止

SFO サーバは、J2EE サーバと同様の手順で停止します。

J2EE サーバの停止手順については、「付録 E.4(4) J2EE サーバの停止」を参照してください。

(6) CTM デーモンの停止

CTM デーモンの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CTM/bin
```

2. ctmstop コマンドで CTM デーモンを停止します。
Solaris の場合、コマンドの実行には、root 権限が必要です。

```
# ctmstop
```

なお、CTM デーモンを停止しても、CTM デーモンの停止処理が完了しない場合があります。この場合、CTM デーモンを強制的に停止する必要があります。CTM デーモンを

強制停止する場合、ctmstop コマンドに -CTMForce オプションを指定して実行します。

(7) CORBA ネーミングサービスの停止

CORBA ネーミングサービスの停止手順は、CORBA ネーミングサービスの起動方法によって異なります。

- J2EE サーバ起動時にインプロセスで起動している場合
J2EE サーバ停止時に CORBA ネーミングサービスも自動停止します。
- フォアグラウンドで手動起動している場合
[Ctrl] + [C] キーで停止します。
- バックグラウンドで手動起動している場合
CORBA ネーミングサービスのプロセス ID を特定して、kill コマンドで停止します。

次に、バックグラウンドで手動起動している場合の停止手順の詳細について説明します。

バックグラウンドで手動起動している場合、ほかの Java プロセスと区別がつかなくなり、どれが CORBA ネーミングサービスなのかわからなくなることがあります。このため、ほかに起動している Java プロセスの有無によって CORBA ネーミングサービスのプロセス ID の取得方法が異なります。なお、AIX の場合は、ps -ef コマンドを実行するとプロセスの引数が他プラットフォームよりも長めに表示されるため、CORBA ネーミングサービスを識別できます。

ほかに起動している Java プロセスがないとき

1. CORBA ネーミングサービスのプロセス ID を取得します。

```
# ps -ef | grep java
```

2. 取得したプロセス ID を指定して、CORBA ネーミングサービスを停止します。

```
# kill <CORBAネーミングサービスのプロセスID>
```

ほかに起動している Java プロセスがあるとき

1. CORBA ネーミングサービスのプロセス ID を取得します。

CORBA ネーミングサービスの起動用シェルスクリプトを使用すると、カレントワーキングディレクトリに生成される namesv_pid ファイルに CORBA ネーミングサービスのプロセス ID を出力できます。

CORBA ネーミングサービスの起動用シェルスクリプトの例を次に示します。

```
#!/bin/sh
export VBROKER_ADM=/opt/Cosminexus/TPB/adm
export SHLIB_PATH="${SHLIB_PATH}:/opt/Cosminexus/TPB/lib"

# start name server process
exec /opt/Cosminexus/TPB/bin/nameserv ¥
-J-Dvbroker.agent.enableLocator=false ¥
-J-Djava.security.policy==/opt/Cosminexus/CC/server/sysconf/cli.policy ¥
-J-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=900 &

# save background java process pid
echo $! > ./namesv_pid
```

なお、下線の部分は、OS ごとに変数名が異なります。AIX の場合は「LIBPATH」、HP-UX (IPF)、Linux および Solaris の場合は「LD_LIBRARY_PATH」となります。

2. 取得したプロセス ID を指定して、CORBA ネーミングサービスを停止します。

```
# kill `cat namesv_pid`
```

(8) CTM ドメインマネージャの停止

CTM ドメインマネージャの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CTM/bin
```

2. ctmdmstop コマンドで CTM ドメインマネージャを停止します。
Solaris の場合、コマンドの実行には、root 権限が必要です。

```
# ctmdmstop
```

なお、CTM ドメインマネージャを停止しても、CTM ドメインマネージャの停止処理が完了しない場合があります。この場合、CTM ドメインマネージャを強制的に停止する必要があります。CTM ドメインマネージャを強制停止する場合、ctmdmstop コマンドに -CTMForce オプションを指定して実行します。

(9) スマートエージェントの停止

スマートエージェントの停止手順を次に示します。

スマートエージェントの停止方法は、起動方法によって異なります。

フォアグラウンドで手動起動している場合

[Ctrl] + [C] キーで停止します。

バックグラウンドで手動起動している場合

次の手順でスマートエージェントを停止します。

1. スマートエージェントのプロセス ID を取得します。

```
# ps -ef | grep osagent
```

2. 取得したプロセス ID を指定して、スマートエージェントを停止します。

```
# kill <スマートエージェントのプロセスID>
```

(10) PRF デーモンの停止

PRF デーモンの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/PRF/bin
```

2. cprfstop コマンドで PRF デーモンを停止します。
Solaris の場合、コマンドの実行には、root 権限が必要です。

```
# cprfstop
```

なお、PRF デーモンを停止しても、PRF デーモンの停止処理が完了しない場合があります。この場合、PRF デーモンを強制的に停止する必要があります。PRF デーモンを強制停止する場合、cprfstop コマンドに -Force オプションを指定して実行します。

付録 F バッチアプリケーションを実行するシステムの構築と削除

ここでは、バッチアプリケーションを実行するシステムを、ファイル編集とコマンド実行で構築、および削除する手順について説明します。

付録 F.1 システム構築の流れ

バッチアプリケーションを実行するシステムをファイル編集とコマンド実行で構築する場合、アプリケーションサーバが提供するコマンド、ユーザ定義ファイルなどを使用してシステムの動作環境を設定します。

(1) システムの構築で使用するファイル

バッチアプリケーションを実行するシステムを構築する場合に使用する主なファイルを次に示します。各ファイルの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編(サーバ定義)」を参照してください。

! 注意事項

ファイルを変更する場合は、バッチサーバを停止してください。バッチサーバの稼働中にファイルの内容を変更した場合の動作は保証されません。

usrconf.properties (バッチサーバ用ユーザプロパティファイル)

バッチサーバのユーザプロパティを指定するファイルです。バッチサーバが使用するネーミングサービス、コネクション、コンテナ、ログ、セキュリティ、CTMの使用などを指定します。

ファイルの格納場所を次に示します。

- Windows の場合
<Cosminexus のインストールディレクトリ>%CC%\server\usrconf\%ejb%<サーバ名称>\usrconf.properties
- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/usrconf.properties

usrconf.cfg (バッチサーバ用オプション定義ファイル)

バッチサーバの JavaVM の起動オプションを指定するファイルです。

ファイルの格納場所を次に示します。

- Windows の場合
<Cosminexus のインストールディレクトリ>%CC%\server\usrconf\%ejb%<サーバ名称>\usrconf.cfg
- UNIX の場合
/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称>/usrconf.cfg

server.policy (バッチサーバ用セキュリティポリシーファイル)

バッチサーバを実行する JavaVM のセキュリティポリシーを指定するファイルです。ファイルの格納場所を次に示します。

- Windows の場合

<Cosminexus のインストールディレクトリ >%CC%\server\usrconf\ejb\<サーバ名称 >%server.policy

- UNIX の場合

/opt/Cosminexus/CC/server/usrconf/ejb/<サーバ名称 >/server.policy

(2) バッチアプリケーションを実行するシステムを構築する流れ

ファイル編集とコマンド実行で、バッチアプリケーションを実行するシステムを構築する場合の、システム構築の流れについて説明します。

1. バッチサーバをセットアップします。

詳細については、「付録 F.2 バッチサーバのセットアップ」を参照してください。

2. バッチサーバの動作設定をします。

使用する機能の動作を変更したり、特定の機能を使用したりする場合には、アプリケーションサーバが提供するユーザ定義ファイルやコマンドを使用して、バッチサーバの動作を設定、変更します。ただし、トランザクションは必ずローカルトランザクションに設定してください。

バッチサーバの動作を設定、変更できる機能の例を次に示します。

- ネーミングサービス
- JavaVM の起動オプション
- コンテナ拡張ライブラリ

アプリケーションサーバが提供する主な機能の設定内容については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 拡張編」のバッチサーバの動作設定に関する説明を参照してください。

3. ジョブスケジューリング機能を使用する場合の設定をします。

バッチアプリケーションの実行をスケジューリングする場合にだけ必要な設定です。スケジューリング機能は CTM を使用します。CTM を使用するための動作設定は次のファイルで指定します。

- usrconf.properties (バッチサーバ用ユーザプロパティファイル) の ejbserver.ctm.enabled プロパティ
- バッチアプリケーションで使用するコマンド (cjexecjob, cjkilljob, cjlistjob) の usrconf.cfg (バッチアプリケーション用オプション定義ファイル) の batch.ctm.enabled オプション

詳細については、マニュアル「Cosminexus アプリケーションサーバ リファレンス 定義編 (サーバ定義)」を参照してください。

4. トラブルシューティングの資料取得の設定をします。

デフォルトの設定のままで利用する場合は、設定不要です。ただし、次の資料は、事

前に取得のための設定をしておかないと、取得できません。これらの資料はトラブルシューティングで必要となるため、取得することをお勧めします。

- OSの統計情報，ユーザダンプ（Windowsの場合）またはcoreダンプ（UNIXの場合）
- JavaVMのガーベージコレクションのログ

これらの資料を取得する場合や、ほかの資料の取得設定を変更したい場合に、トラブルシューティングの資料取得の設定をしてください。トラブルシューティングに必要な資料の詳細と、システム構築時の設定の要否については「付録 F.1(3) トラブルシューティングに必要な資料」を参照してください。

5. リソースの設定をします。

サーバ管理コマンドを使用して、リソースアダプタのプロパティを定義し、バッチサーバにインポートします。設定方法の詳細については、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」のリソースの設定に関する説明を参照してください。

6. システムを起動して、システムの動作を確認します。

システムの起動および停止については、「付録 G バッチアプリケーションを実行するシステムの起動と停止（Windowsの場合）」、または「付録 H バッチアプリケーションを実行するシステムの起動と停止（UNIXの場合）」を参照してください。

！ 注意事項

各ファイルでデフォルトの設定を変更する場合、簡易構築定義ファイルを使用して設定しているときは、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」のJ2EEサーバの動作設定についての説明を参照してください。

(3) トラブルシューティングに必要な資料

トラブルシューティングに必要な資料と、システム構築時の設定の要否について次の表に示します。なお、設定方法の詳細については、マニュアル「Cosminexus アプリケーションサーバ機能解説 保守 / 移行 / 互換編」のトラブルシューティングの資料取得の設定に関する説明を参照してください。

表 F-1 トラブルシューティングの資料の種類と設定要否（バッチアプリケーションを実行するシステムの場合）

資料の種類	設定の要否
snapshot ログ	1
性能解析トレースファイル	x 2
バッチサーバのログ	
アプリケーションのユーザログ	
稼働情報ファイル	

資料の種類	設定の要否
リソースアダプタのログ	
Cosminexus TPBroker のトレースファイル	
Cosminexus DABroker Library のログ	
OS の統計情報	
ユーザダンプ	
core ダンプ	
JavaVM の資料	
サーバ管理コマンドのログ	

(凡例)

- : 資料を取得する場合には、設定が必要。
- : デフォルトの資料取得の設定を変更する場合だけ、設定が必要。
- × : 設定は不要。

注 1

snapshotlog コマンドを使用して snapshot ログを取得します。この場合、あらかじめ設定しておいたリストファイルを、コマンド実行時に引数として指定します。snapshotlog コマンド、およびデフォルトの設定で収集できる snapshot ログについては、「11.2 ログの運用で使用するコマンドの詳細」の「snapshotlog (snapshot ログの収集)」を参照してください。なお、snapshot ログの収集先を変更する場合には、snapshot ログ収集対象定義ファイルを編集します。snapshot ログ収集対象定義ファイルについては、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編 (サーバ定義)」を参照してください。

注 2

日常的なシステム運用の作業で、cprfstart コマンドを実行してパフォーマンストレーサを起動する場合、PRF トレースの取得レベルやファイルの面数を指定できます。性能解析トレーサファイルの取得については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

表に示したログは、snapshot ログで一括収集できます。ただし、CosminexusTPBroker のトレースファイルについては、収集できるものと収集できないものが混在しています。このほかにも、snapshot ログのデフォルトの設定で取得できないログについては、取得のための設定をしたり、snapshot ログの収集先に追加したりする必要があります。

なお、次のログはログ出力先を変更できません。

- セットアップ時に作成される install.log、移行コマンドなどのログファイル
- 次のディレクトリに出力される Java のスレッドダンプファイル
 - Windows の場合
 - < 作業ディレクトリ >¥ejb¥< サーバ名称 >
 - UNIX の場合
 - < 作業ディレクトリ >/ejb/< サーバ名称 >

ログの種類やデフォルト値、チャンネル名、取得できるログの詳細や取得方法については、マニュアル「Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編」を参照してください。

付録 F.2 バッチサーバのセットアップ

アプリケーションサーバのシステムを構成するバッチサーバのセットアップについて説明します。

デフォルトの設定では、バッチサーバ名はホスト名です。また、同じホスト内で複数のバッチサーバを起動する場合には、それぞれのバッチサーバに別のサーバ名を割り当ててください。サーバ名を割り当てるときの注意事項については、「付録 F.2 (4) 複数のバッチサーバを起動する際の注意事項」を参照してください。

バッチサーバのセットアップの手順について説明します。セットアップに関する注意事項については、「付録 F.2 (3) セットアップに関する注意事項」を参照して、必要に応じて対応してください。

(1) Windows の場合

Windows の場合のバッチサーバのセットアップ手順を次に示します。

1. 次のコマンドでカレントディレクトリを移動します。
"cd <Cosminexus のインストールディレクトリ>%CC%server%bin"

2. cjsetup コマンドでバッチサーバをセットアップします。

"cjsetup -batch <サーバ名>"

バッチサーバのセットアップの例を次に示します。この例では、Cosminexus のインストールディレクトリは「C:%Program Files%Hitachi%Cosminexus」、サーバ名は「MyServer」です。

```
C:%>cd C:%Program Files%Hitachi%Cosminexus%CC%server%bin
C:%Program Files%Hitachi%Cosminexus%CC%server%bin>cjsetup -batch MyServer
```

(2) UNIX の場合

UNIX の場合のバッチサーバのセットアップ手順を次に示します。

1. cjsetup コマンドでバッチサーバをセットアップします。コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

/opt/Cosminexus/CC/server/bin/cjsetup -batch <サーバ名>

バッチサーバのセットアップの例を次に示します。この例では、サーバ名は「MyServer」です。

```
/opt/Cosminexus/CC/server/bin/cjsetup -batch MyServer
```

(3) セットアップに関する注意事項

バッチサーバのセットアップは、1 回だけ行ってください。セットアップ済みのバッチサーバを指定するとエラーメッセージが出力されます。

バッチサーバをセットアップして、バッチサーバをカスタマイズしない場合、バッチサーバはホスト名「localhost」、ポート番号「900」の CORBA ネーミングサービスをインプロセスで自動起動して使用します。使用する CORBA ネーミングサービスを変更する場合は、usrconf.properties の ejbserver.naming.host キーと ejbserver.naming.port キーで変更してください。

バッチサーバを開始する前に、環境変数「OSAGENT_PORT」の値と usrconf.properties の vbroker.agent.port キーの値が一致しているかどうかを確認してください。

サーバ名称に使用できる文字については、cjsetup コマンドの説明を参照してください。cjsetup コマンドについては、「3.2 バッチサーバを操作するコマンド」を参照してください。

サーバ名称は、作業ディレクトリ中のディレクトリ名として用いられます。作業ディレクトリ下のパス長がプラットフォームの上限に達しないようにサーバ名称を指定してください。作業ディレクトリのパス長の見積もりについては、マニュアル「Cosminexus アプリケーションサーバシステム構築・運用ガイド」のバッチサーバの作業ディレクトリについての説明を参照してください。

(4) 複数のバッチサーバを起動する場合の注意事項

バッチサーバはサーバ名称によって識別されます。インストールされたホストのホスト名がデフォルトのサーバ名称として設定されます。このため、同じホスト内で複数のバッチサーバを起動する場合、それぞれのバッチサーバにサーバ名称を割り当て、バッチサーバをセットアップします。

バッチサーバにサーバ名称を割り当てる場合、同じホスト内では、同じサーバ名称のバッチサーバを複数回セットアップしてはいけません。同じサーバ名称で再セットアップすると、エラーになります。

バッチサーバと接続する CORBA ネーミングサービスのホスト名およびポート番号に関する注意事項を次に示します。

ホスト名およびポート番号の設定

同一ホストで複数のバッチサーバを同時に起動する場合、CORBA ネーミングサービスとの接続に使用するホスト名およびポート番号が両方とも重ならないようにしてください。ホスト名およびポート番号は、usrconf.properties で設定します。

また、usrconf.properties の webserver.connector.ajp13.port キーおよび

ejbserver.http.port キーで設定されているポート番号には、同時に起動するバッチサーバ間で異なる値を使用するように設定してください。

付録 F.3 システムの削除の流れ

バッチアプリケーションを実行するシステムをファイル編集とコマンド実行で削除する場合、アプリケーションサーバが提供するコマンドを使用して、バッチサーバをアンセットアップし、システムを削除します。

ファイル編集とコマンド実行で、バッチアプリケーションを実行するシステムを削除する場合の流れについて説明します。

1. システムを停止します。
システムの停止手順および停止方法については、「付録 G バッチアプリケーションを実行するシステムの起動と停止 (Windows の場合)」, または「付録 H バッチアプリケーションを実行するシステムの起動と停止 (UNIX の場合)」を参照してください。
2. バッチサーバをアンセットアップします。
詳細については、「付録 F.4 バッチサーバのアンセットアップ」を参照してください。

付録 F.4 バッチサーバのアンセットアップ

アプリケーションサーバのシステムを構成するバッチサーバのアンセットアップについて説明します。バッチサーバのアンセットアップの手順について、OS ごとに説明します。

(1) Windows の場合

Windows の場合のバッチサーバのアンセットアップ手順を次に示します。

1. 次のコマンドでカレントディレクトリを移動します。
"cd <Cosminexus のインストールディレクトリ>%CC%server%bin"
2. cjsetup コマンドで引数に -d を指定して、バッチサーバをアンセットアップします。
"cjsetup -d <サーバ名>"
バッチサーバのアンセットアップの例を次に示します。この例では、Cosminexus のインストールディレクトリは「C:\Program Files\Hitachi\Cosminexus」、サーバ名は「MyServer」です。

```
C:>cd C:\Program Files\Hitachi\Cosminexus\CC\server\bin
C:\Program Files\Hitachi\Cosminexus\CC\server\bin>cjsetup -d MyServer
```

(2) UNIX の場合

UNIX の場合のバッチサーバのアンセットアップ手順を次に示します。

1. cjsetup コマンドで引数に -d を指定して、バッチサーバをアンセットアップします。コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

`/opt/Cosminexus/CC/server/bin/cjsetup -d <サーバ名>`

バッチサーバのアンセットアップの例を次に示します。この例では、サーバ名は「MyServer」です。

```
/opt/Cosminexus/CC/server/bin/cjsetup -d MyServer
```

付録 G バッチアプリケーションを実行するシステムの起動と停止 (Windows の場合)

ここでは、バッチアプリケーションを実行するシステムの起動と停止の手順について説明します。

付録 G.1 システムの起動手順

システムの起動手順について説明します。

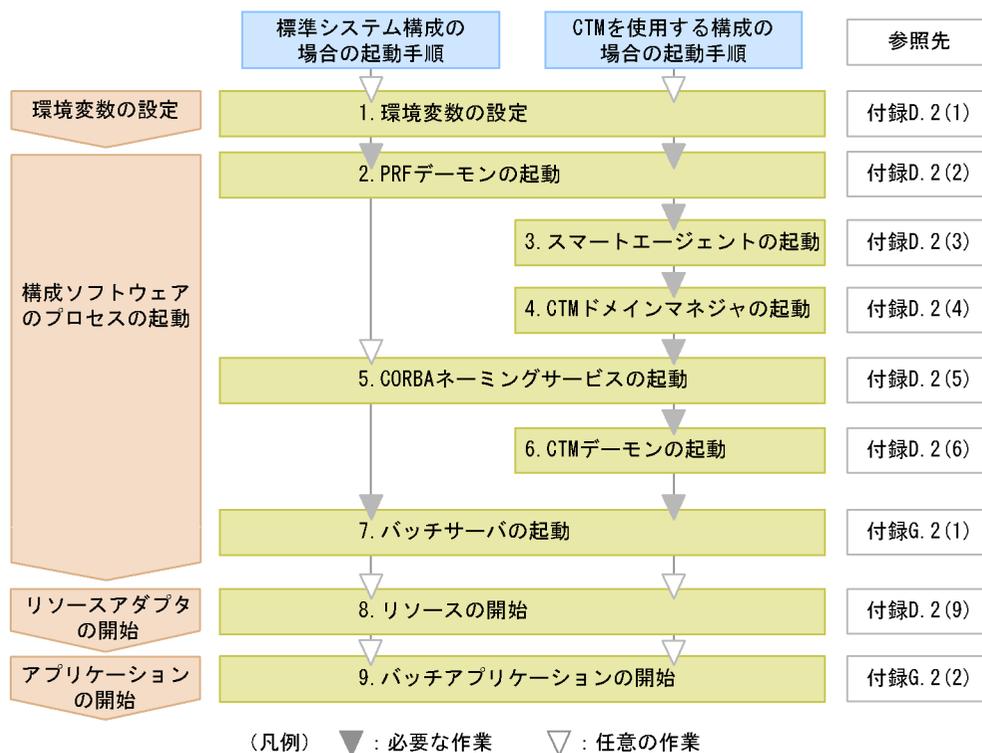
なお、データベースは起動済みであることを前提に説明します。データベースを使用している場合は、必要に応じてあらかじめ起動しておいてください。

(1) アプリケーションサーバの起動の流れ

システムで使用するアプリケーションサーバの起動の流れについて説明します。

システムの起動の流れを次の図に示します。

図 G-1 システムの起動の流れ (Windows の場合)



図中の 1. ~ 9. について説明します。

1. 環境変数の設定
環境変数を設定します。環境変数の設定方法については、「付録 D.2(1) システム起動時の環境変数の設定」を参照してください。
2. PRF デーモンの起動
cprfstart コマンドで PRF デーモンを起動します。起動方法については、「付録 D.2(2) PRF デーモンの起動」を参照してください。
3. スマートエージェントの起動
CTM を使用しない場合、この操作は不要です。
osagent コマンドでスマートエージェントを起動します。起動方法については、「付録 D.2(3) スマートエージェントの起動」を参照してください。
4. CTM ドメインマネージャの起動
CTM を使用しない場合、この操作は不要です。
ctmdmstart コマンドで CTM ドメインマネージャを起動します。起動方法については、「付録 D.2(4) CTM ドメインマネージャの起動」を参照してください。
5. CORBA ネーミングサービスの起動
CTM を使用しない場合、この操作は不要です。
CTM を使用する場合、nameserv コマンドで CTM 用の CORBA ネーミングを起動します。起動方法については、「付録 D.2(5) CORBA ネーミングサービスの起動」を参照してください。
6. CTM デーモンの起動
CTM を使用しない場合、この操作は不要です。
ctmstart コマンドで CTM デーモンを起動します。起動方法については、「付録 D.2(6) CTM デーモンの起動」を参照してください。
7. バッチサーバの起動
cjstartsv コマンドでバッチサーバを起動します。起動方法については、「付録 G.2(1) バッチサーバの起動」を参照してください。
8. リソースの開始
リソースを使用しない場合、この操作は不要です。
サーバ管理コマンド (cjstartrar) でリソースを開始します。開始方法については、「付録 D.2(9) リソースの開始」を参照してください。
9. バッチアプリケーションの実行
cjexecjob コマンドでバッチアプリケーションを実行します。
実行方法については、「付録 G.2(2) バッチアプリケーションの実行」を参照してください。

付録 G.2 システムの起動方法

ここでは、バッチサーバの起動方法、およびバッチアプリケーションの実行方法につい

て説明します。

(1) バッチサーバの起動

バッチサーバの起動方法について説明します。

標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合の、バッチサーバの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%\server\bin"
```

2. cjstartsv コマンドでバッチサーバを開始します。
コマンドの実行には、Administrator 権限が必要です。

```
cjstartsv <サーバ名称> -nosecurity
```

(2) バッチアプリケーションの実行

バッチアプリケーションの実行方法について説明します。

バッチアプリケーションは、cjexecjob コマンドを使用して実行します。なお、cjexecjob コマンドの実行には、Administrator 権限が必要です。Windows Vista の場合は、管理者特権が必要です。

バッチアプリケーションの実行手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%\batch\bin"
```

2. cjexecjob コマンドでバッチアプリケーションを実行します。
コマンドの実行には、Administrator 権限が必要です。
バッチアプリケーションがジョブスケジューリング機能を使用する場合と使用しない場合とでは接続サーバの指定方法が異なります。
ジョブスケジューリング機能を使用しない場合
ジョブの実行ができるバッチサーバを指定します。

```
cjexecjob <サーバ名称> <Javaアプリケーションクラス名>
```

ジョブスケジューリング機能を使用する場合
ジョブスケジューリングのグループを指定します。

```
cjexecjob [<スケジュールグループ名>] [-jobID <ジョブID>] <Javaアプリケーションクラス名>
```

付録 G.3 システムの停止手順

ここでは、バッチアプリケーション、およびバッチサーバなどの停止の手順について説明します。

(1) システムの停止の流れ

システムの停止手順は、起動手順と逆の順序になります。データベースを使用している場合は、必要に応じて最後に停止してください。

次に示す順にシステムを停止してください。

バッチアプリケーションの停止

バッチアプリケーションが停止していない場合に、`ckilljob` コマンドでバッチアプリケーションを停止します。停止の方法については、「付録 G.4(1) バッチアプリケーションの停止」を参照してください。

リソースの停止

リソースを使用していない場合、この操作は不要です。

サーバ管理コマンド (`cjstoprar`) でリソースを停止します。停止方法については、「付録 D.4(3) リソースの停止」を参照してください。

バッチサーバの停止

`cjstopsv` コマンドでバッチサーバを停止します。停止方法については、「付録 G.4(2) バッチサーバの停止」を参照してください。

CTM デーモンの停止

CTM を使用しない場合、この操作は不要です。

`ctmstop` コマンドで CTM デーモンを停止します。停止方法については、「付録 D.4(6) CTM デーモンの停止」を参照してください。

CORBA ネーミングサービスの停止

[Ctrl] + [C] で CORBA ネーミングサービスを停止します。

CTM ドメインマネージャの停止

CTM を使用しない場合、この操作は不要です。

`ctmdmstop` コマンドで CTM ドメインを停止します。停止方法については、「付録 D.4(8) CTM ドメインマネージャの停止」を参照してください。

スマートエージェントの停止

スマートエージェントのアイコンを右クリックして `Exit` を選択してスマートエージェントを停止します。

PRF デーモンの停止

eprfstop コマンドで PRF デーモンを停止します。停止方法については、「付録 D.4(10) PRF デーモンの停止」を参照してください。

付録 G.4 システムの停止方法

ここでは、バッチアプリケーションおよびバッチサーバの停止の方法について説明します。

(1) バッチアプリケーションの停止

バッチアプリケーションの停止の方法について説明します。

バッチアプリケーションが停止していない場合に、cjkiljob コマンドを使用してバッチアプリケーションを停止します。なお、cjkiljob コマンドの実行には、Administrator 権限が必要です。また、Windows Vista の場合は、管理者特権が必要です。

cjkiljob コマンドは、メソッドキャンセルまたは CTM によりバッチアプリケーションを停止します。停止に失敗した場合は、バッチサーバを強制停止します。

バッチアプリケーションの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%batch%bin"
```

2. cjkiljob コマンドでバッチサーバを停止します。

コマンドの実行には、Administrator 権限が必要です。

バッチアプリケーションの停止には、サーバ名称を指定する場合とジョブ ID を指定する場合があります。

サーバ名称を指定する場合

メソッドキャンセルによってバッチアプリケーションを停止します。

```
cjkiljob <サーバ名称>
```

ジョブ ID を指定する場合

CTM によってバッチアプリケーションを停止します。

ジョブスケジューリング機能を使用している場合にだけ使用できます。

```
cjkiljob -jobID <ジョブID>
```

(2) バッチサーバの停止

バッチサーバの停止の手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd "<Cosminexusのインストールディレクトリ>%CC%server%bin"
```

2. cjstopsv コマンドでバッチサーバを停止します。
コマンドの実行には、Administrator 権限が必要です。

```
cjstopsv <サーバ名称>
```

cjstopsv コマンドの実行例を次に示します。この例では、Cosminexus のインストールディレクトリは「C:\Program Files\Hitachi\Cosminexus」、サーバ名は「MyServer」です。

```
C:>cd "C:\Program Files\HITACHI\Cosminexus\CC\server\bin"
C:\Program Files\HITACHI\Cosminexus\CC\server\bin>cjstopsv MyServer
C:\Program Files\HITACHI\Cosminexus\CC\server\bin>
```

なお、バッチサーバでの処理が終了しないなどの原因で、バッチサーバの停止処理が完了しない場合があります。この場合、バッチサーバを強制的に停止する必要があります。バッチサーバを強制停止する場合、cjstopsv コマンドに -f オプションまたは -fd オプションを指定して実行します。-fd オプションを指定すると、バッチサーバの障害情報を取得したあと、バッチサーバを強制停止します。

付録 H バッチアプリケーションを実行するシステムの起動と停止 (UNIX の場合)

ここでは、バッチアプリケーションを実行するシステムの起動と停止の手順について説明します。

付録 H.1 システムの起動手順

システムの起動手順について説明します。

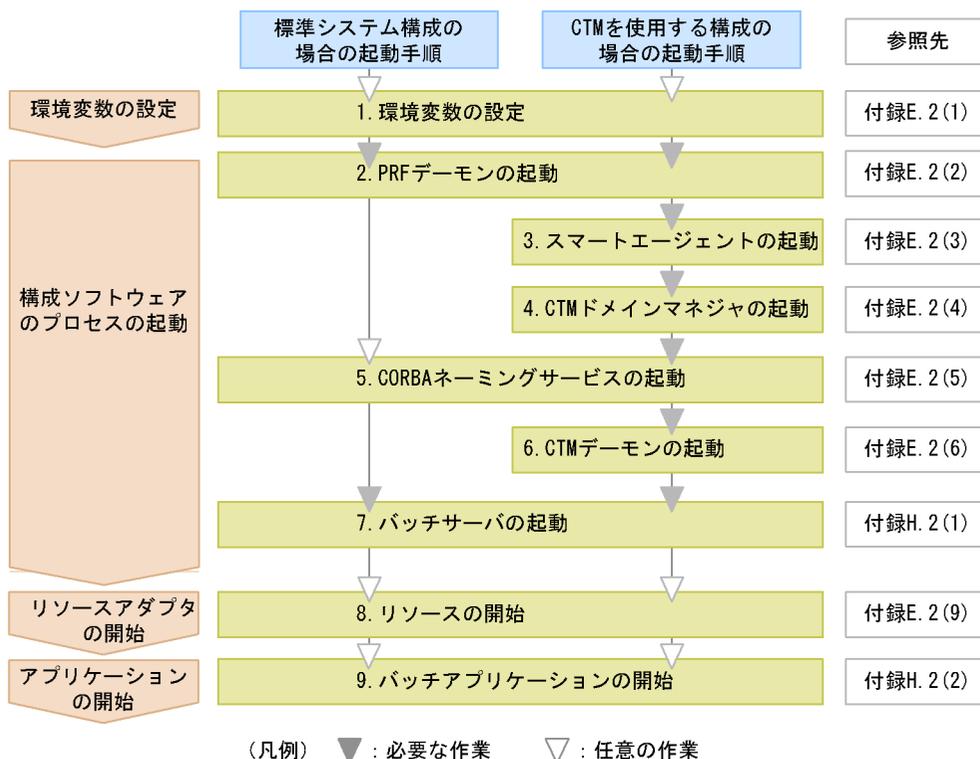
なお、データベースは起動済みであることを前提に説明します。データベースを使用している場合は、必要に応じてあらかじめ起動しておいてください。

(1) アプリケーションサーバの起動の流れ

システムで使用するアプリケーションサーバの起動の流れについて説明します。

アプリケーションサーバの起動の流れを次の図に示します。

図 H-1 アプリケーションサーバの起動の流れ (UNIX の場合)



図中の 1. ~ 9. について説明します。

1. 環境変数の設定
環境変数を設定します。環境変数の設定方法については、「付録 E.2(1) システム起動時の環境変数の設定」を参照してください。
2. PRF デーモンの起動
cprfstart コマンドで PRF デーモンを起動します。起動方法については、「付録 E.2(2) PRF デーモンの起動」を参照してください。
3. スマートエージェントの起動
CTM を使用しない場合、この操作は不要です。
osagent コマンドでスマートエージェントを起動します。起動方法については、「付録 E.2(3) スマートエージェントの起動」を参照してください。
4. CTM ドメインマネージャの起動
CTM を使用しない場合、この操作は不要です。
ctmdmstart コマンドで CTM ドメインマネージャを起動します。起動方法については、「付録 E.2(4) CTM ドメインマネージャの起動」を参照してください。
5. CORBA ネーミングサービスの起動
CTM を使用しない場合、この操作は不要です。
CTM を使用する場合、nameserv コマンドで CTM 用の CORBA ネーミングを起動します。起動方法については、「付録 E.2(5) CORBA ネーミングサービスの起動」を参照してください。
6. CTM デーモンの起動
CTM を使用しない場合、この操作は不要です。
ctmstart コマンドで CTM デーモンを起動します。起動方法については、「付録 E.2(6) CTM デーモンの起動」を参照してください。
7. バッチサーバの起動
cjstartsv コマンドでバッチサーバを起動します。起動方法については、「付録 H.2(1) バッチサーバの起動」を参照してください。
8. リソースの開始
リソースを使用しない場合、この操作は不要です。
サーバ管理コマンドでリソースを開始します。開始方法については、「付録 E.2(9) リソースの開始」を参照してください。
9. バッチアプリケーションの実行
cjexecjob コマンドでバッチアプリケーションを開始します。開始方法については、「付録 H.2(2) バッチアプリケーションの実行」を参照してください。

付録 H.2 システムの起動方法

ここでは、バッチサーバの起動方法、およびバッチアプリケーションの実行方法について説明します。

(1) バッチサーバの起動

バッチサーバの起動方法について説明します。

バッチサーバ起動時は次の点に留意してください。

- cjstartsv コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。
- root 権限を持たないユーザを Component Container 管理者に設定している場合、CORBA ネーミングサービスを起動するときは、バッチサーバ用の CORBA ネーミングサービスのポート番号に 1024 以上の値を指定しておく必要があります。1024 以上の値が指定されていない場合、エラーとなり、バッチサーバは起動できません。
- バッチサーバ起動時に KDJE47500-E というエラーメッセージが出力された場合、利用しようとした CORBA ネーミングサービスがすでに起動されているため、バッチサーバの起動処理に失敗します。すでに起動している CORBA ネーミングサービスを停止して、バッチサーバを起動してください。

標準システム構成で CORBA ネーミングサービスをインプロセスで起動する場合の、バッチサーバの起動手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. cjstartsv コマンドでバッチサーバを開始します。

コマンドの実行には、root 権限 (Component Container 管理者を設定していない場合)、または Component Container 管理者の権限 (Component Container 管理者を設定している場合) が必要です。

```
# cjstartsv <サーバ名称> -nosecurity
```

(2) バッチアプリケーションの実行

バッチアプリケーションの実行方法について説明します。

バッチアプリケーションは、サーバ管理コマンドを使用して実行します。なお、サーバ管理コマンドの実行には、root 権限または Component Container 管理者の権限が必要です。

バッチアプリケーションの実行手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
cd /opt/Cosminexus/CC/batch/bin/cjexecjob
```

2. cjexecjob コマンドでバッチアプリケーションを実行します。

コマンドの実行には、root 権限または Component Container 管理者の権限が必要です。

バッチアプリケーションがジョブスケジューリング機能を使用する場合と使用しない場合とでは接続サーバの指定方法が異なります。

ジョブスケジューリング機能を使用しない場合
ジョブの実行ができるバッチサーバを指定します。

```
cjexecjob.exe <サーバ名称> <Javaアプリケーションクラス名>
```

ジョブスケジューリング機能を使用する場合
ジョブスケジューリングのグループを指定します。

```
cjexecjob [<スケジュールグループ名>] [-jobID <ジョブID>] <Javaアプリケーションクラス名>
```

付録 H.3 システムの停止手順

ここでは、バッチアプリケーション、およびバッチサーバなどの停止の手順について説明します。

(1) システムの停止の流れ

システムの停止手順は、起動手順と逆の順序になります。データベースを使用している場合は、必要に応じて最後に停止してください。

次に示す順にシステムを停止してください。

バッチアプリケーションの停止

バッチアプリケーションが停止していない場合に、cjkilljob コマンドでバッチアプリケーションを停止します。停止方法については、「付録 H.4(1) バッチアプリケーションの停止」を参照してください。

リソースの停止

リソースを使用していない場合、この操作は不要です。

サーバ管理コマンド (cjstoprar) でリソースを停止します。停止方法については、「付録 E.4(3) リソースの停止」を参照してください。

バッチサーバの停止

cjstopsv コマンドでバッチサーバを停止します。停止方法については、「付録 H.4(2) バッチサーバの停止」を参照してください。

CTM デーモンの停止

CTM を使用しない場合、この操作は不要です。

ctmstop コマンドで CTM デーモンを停止します。停止方法については、「付録 E.4(6)

CTM デーモンの停止」を参照してください。

CORBA ネーミングサービスの停止

[Ctrl] + [C] で CORBA ネーミングサービスを停止します。

CTM ドメインマネージャの停止

CTM を使用しない場合、この操作は不要です。

ctmdmstop コマンドで CTM ドメインを停止します。停止方法については、「付録 E.4(8) CTM ドメインマネージャの停止」を参照してください。

スマートエージェントの停止

[Ctrl] + [C] でスマートエージェントを停止します。

PRF デーモンの停止

cprfstop コマンドで PRF デーモンを停止します。停止方法については、「付録 E.4(10) PRF デーモンの停止」を参照してください。

付録 H.4 システムの停止方法

ここでは、バッチアプリケーションおよびバッチサーバの停止方法について説明します。

(1) バッチアプリケーションの停止

バッチアプリケーションの停止方法について説明します。

バッチアプリケーションが停止していない場合に、cckilljob コマンドを使用してバッチアプリケーションを停止します。なお、cckilljob コマンドの実行には、root 権限または Component Container 管理者の権限が必要です。

cckilljob コマンドは、メソッドキャンセルまたは CTM によりバッチアプリケーションを停止します。停止に失敗した場合は、バッチサーバを強制停止します。

バッチアプリケーションの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/batch/bin
```

2. cckilljob コマンドでバッチアプリケーションを停止します。

コマンドの実行には、root 権限または Component Container 管理者の権限が必要です。

バッチアプリケーションの停止には、サーバ名称を指定する場合とジョブ ID を指定する場合があります。

サーバ名称を指定する場合

メソッドキャンセルによってバッチアプリケーションを停止します。

```
cjkilljob <サーバ名称>
```

ジョブ ID を指定する場合

CTM によってバッチアプリケーションを停止します。

ジョブスケジューリング機能を使用している場合にだけ使用できます。

```
cjkilljob -jobID <ジョブID>
```

(2) バッチサーバの停止

バッチサーバの停止手順を次に示します。

1. cd コマンドでカレントディレクトリを移動します。

```
# cd /opt/Cosminexus/CC/server/bin
```

2. cjstopsv コマンドでバッチサーバを停止します。

コマンドの実行には、root 権限または Component Container 管理者の権限が必要です。

```
# cjstopsv <サーバ名称>
```

cjstopsv コマンドの実行例を次に示します。この例では、サーバ名は「MyServer」です。

```
# cd /opt/Cosminexus/CC/server/bin
# cjstopsv MyServer
#
```

なお、バッチサーバでの処理が終了しないなどの原因で、バッチサーバの停止処理が完了しない場合があります。この場合、バッチサーバを強制的に停止する必要があります。バッチサーバを強制停止する場合、cjstopsv コマンドに `-f` オプションまたは `-fd` オプションを指定して実行します。`-fd` オプションを指定すると、バッチサーバの障害情報を取得したあと、バッチサーバを強制停止します。

付録 I Smart Composer 機能で使用できる抽象パラメタ

Smart Composer 機能では、互いに関連があるパラメタを一つにまとめたパラメタとして提供しています。これを**抽象パラメタ**といいます。抽象パラメタとは、個々の製品のそれぞれ具体的な実パラメタではなく、システム設計の見地からシステムの性能要件として定義されるパラメタのことをいいます。Smart Composer 機能では、次の抽象パラメタを使用します。

リクエスト最大同時処理数

一つのサービスユニットでの、リクエストの最大同時処理数です。リクエストとは、クライアントから送られる HTTP リクエストに端を発する、一連の業務処理をいいます。

cmx_trans_param コマンドを実行すると、定義した抽象パラメタの値が、展開規則に従って複数の実パラメタとして展開されます。なお、そのほかの実パラメタについては、パラメタのデフォルト値を使用します。それぞれのパラメタの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編(サーバ定義)」を参照してください。

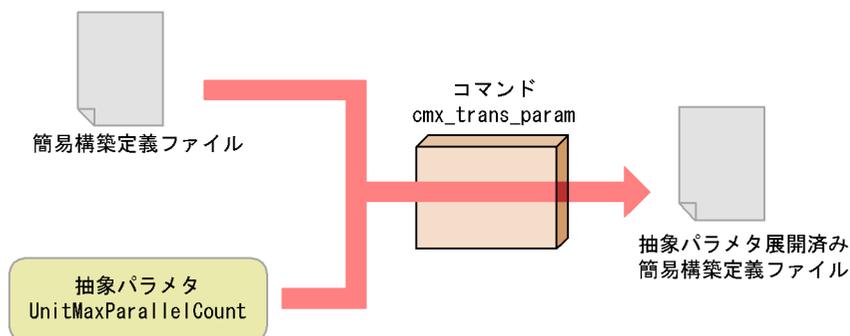
抽象パラメタの展開機能は、物理ティアが combined-tier 構成の場合だけ利用できます。combined-tier 以外の物理ティアの場合は、Smart Composer 機能を利用しない場合の Web システム構築と同様のパラメタ定義設計が必要です。

ここでは、抽象パラメタの展開、および抽象パラメタによって展開される実パラメタについて説明します。

付録 I.1 抽象パラメタの展開

cmx_trans_param コマンドに、簡易構築定義ファイルと抽象パラメタを入力することによって、抽象パラメタから実パラメタに展開された定義ファイル(抽象パラメタ展開済み簡易構築定義ファイル)が作成されます。抽象パラメタ展開のイメージを、次の図に示します。

図 I-1 抽象パラメタ展開のイメージ



なお、抽象パラメタの展開には、抽象パラメタが設計されている必要があります。抽象パラメタを展開する前に、リクエスト最大同時処理数の見積もりをしてください。

次に、リクエスト最大同時処理数を 20 と見積もった場合の抽象パラメタの展開の実行例を示します。

Windows の場合

```
<Cosminexusインストールディレクトリ>%manager%bin%cmx_trans_param -i <任意のディレクトリ>%MyWebSystemDef.xml -p UnitMaxParallelCount=20 -o <任意のディレクトリ>%MyWebSystemDef2.xml
```

UNIX の場合

```
# /opt/Cosminexus/manager/bin/cmx_trans_param -i <任意のディレクトリ>/MyWebSystemDef.xml -p UnitMaxParallelCount=20 -o <任意のディレクトリ>/MyWebSystemDef2.xml
```

cmx_trans_param コマンドの詳細については、「8.3 Smart Composer 機能で使用するコマンドの詳細」の「cmx_trans_param」を参照してください。

付録 I.2 抽象パラメタによって展開される実パラメタ

抽象パラメタであるリクエスト最大同時処理数は、cmx_trans_param コマンドを実行すると、簡易構築定義ファイルの次の七つの実パラメタに値が展開されます。

- ThreadsPerChild
- MaxClients
- StartServers
- HWSKeepStartServers
- KeepAliveTimeout
- webservice.connector.ajp13.backlog

- `webserver.connector.ajp13.max_threads`

また、この抽象パラメタを使用して、Connector 属性ファイルの、次の二つのパラメタの値を決定できます。なお、Connector 属性ファイルとは、リソースアダプタの属性を取得、編集する場合に使用するファイルです。

- `MaxPoolSize`
- `MinPoolSize`

ここでは、簡易構築定義ファイルに展開される実パラメタの展開規則と、Connector 属性ファイルに設定するパラメタの値の設定規則について説明します。

(1) 簡易構築定義ファイルに展開される実パラメタ

リクエスト最大同時処理数の値が、それぞれの実パラメタへ展開される際の規則および実パラメタの説明を、実パラメタごとに説明します。なお、説明中の「設定先のファイル」とは、抽象パラメタ展開済みの簡易構築定義ファイルの情報が最終的に反映される、`Cosminexus` のファイルを指します。

(a) `ThreadsPerChild`

展開規則

$(\text{KeepAliveTimeout} + 1 \text{ リクエストの平均処理時間}) / 1 \text{ リクエストの平均処理時間} \times \text{リクエスト最大同時処理数}$

展開規則は上記のとおりですが、ここでは、1 リクエストの平均処理時間 = `KeepAliveTimeout` として計算されるため、次のような式になります。

リクエスト最大同時処理数 \times 2

実パラメタの説明

Hitachi Web Server で、Web サーバとして起動するスレッド数を指定するパラメタです。指定したスレッド数は Web サーバの最大同時接続数を示します。詳細については、マニュアル「Hitachi Web Server」を参照してください。

設定先のファイル：`httpd.conf` (Hitachi Web Server 定義ファイル)

(b) `MaxClients`

展開規則

$(\text{KeepAliveTimeout} + 1 \text{ リクエストの平均処理時間}) / 1 \text{ リクエストの平均処理時間} \times \text{リクエスト最大同時処理数}$

展開規則は上記のとおりですが、ここでは、1 リクエストの平均処理時間 = `KeepAliveTimeout` として計算されるため、次のような式になります。

リクエスト最大同時処理数 × 2

実パラメタの説明

Hitachi Web Server で、同時に接続できるクライアントの最大数を指定するパラメタです。

設定先のファイル：httpsd.conf (Hitachi Web Server 定義ファイル)

(c) StartServers

展開規則

$(\text{KeepAliveTimeout} + 1 \text{ リクエストの平均処理時間}) / 1 \text{ リクエストの平均処理時間} \times \text{リクエスト最大同時処理数}$

展開規則は上記のとおりですが、ここでは、1 リクエストの平均処理時間 = KeepAliveTimeout として計算されるため、次のような式になります。

リクエスト最大同時処理数 × 2

実パラメタの説明

Hitachi Web Server で、Web サーバ起動時のサーバプロセス数を指定するパラメタです。

詳細については、マニュアル「Hitachi Web Server」を参照してください。

設定先のファイル：httpsd.conf (Hitachi Web Server 定義ファイル)

(d) HWSKeepStartServers

展開規則

「On」が展開されます。

実パラメタの説明

Hitachi Web Server で、サーバプロセスの稼働数を StartServers パラメタに指定した数だけ維持するかどうかを指定するパラメタです。

詳細については、マニュアル「Hitachi Web Server」を参照してください。

設定先のファイル：httpsd.conf (Hitachi Web Server 定義ファイル)

(e) KeepAliveTimeout

展開規則

「3」(秒)が展開されます。

実パラメタの説明

KeepAliveTimeout は、HTTP1.1 の持続型接続をしているときの、コネクションを

持続する時間を指定するパラメタです。

持続型接続を使用する場合 (KeepAlive ディレクティブをオンにする場合), レスポンスの高速化のため, クライアントからのリクエストに対して応答を返したあともコネクションを切断しないで, 同じクライアントからの次のリクエストを待ちます。このため, KeepAliveTimeout でタイムアウト時間を設定します。

詳細については, マニュアル「Hitachi Web Server」を参照してください。

設定先のファイル: httpd.conf (Hitachi Web Server 定義ファイル)

(f) webserver.connector.ajp13.backlog

展開規則

$(\text{KeepAliveTimeout} + 1 \text{ リクエストの平均処理時間}) / 1 \text{ リクエストの平均処理時間} \times \text{リクエスト最大同時処理数}$

展開規則は上記のとおりですが, ここでは, 1 リクエストの平均処理時間 = KeepAliveTimeout として計算されるため, 次のような式になります。

リクエスト最大同時処理数 \times 2

実パラメタの説明

リダイレクタからの接続要求の最大の待ち行列数を指定します。

設定先のファイル: usrconf.properties (J2EE サーバ用ユーザプロパティファイル)

(g) webserver.connector.ajp13.max_threads

展開規則

リクエスト最大同時処理数がそのまま展開されます。

実パラメタの説明

Web コンテナがリクエストを処理する同時実行数を指定します。

設定先のファイル: usrconf.properties (J2EE サーバ用ユーザプロパティファイル)

(2) 抽象パラメタの展開例

cmx_trans_param コマンドを実行すると物理ティアの定義 (<tier> タグの定義) の <configuration> タグに, 次の表に示すパラメタが自動展開されて, 抽象パラメタ展開済み簡易構築定義ファイルが作成されます。

表 I-1 cmx_trans_param コマンドによって自動展開されるパラメタ一覧

logical-server-type	param-name
web-server	ThreadsPerChild
	MaxClients

logical-server-type	param-name
	StartServers
	HWSKeepStartServers
	KeepAliveTimeout
j2ee-server	webserver.connector.ajp13.backlog
	webserver.connector.ajp13.max_threads

なお、パラメタの詳細については、マニュアル「Cosminexus アプリケーションサーバリファレンス 定義編（サーバ定義）」を参照してください。

抽象パラメタ展開済み簡易構築定義ファイルの内容を、次に示します（一部抜粋）。

抽象パラメタ展開済み簡易構築定義ファイル

```
<?xml version="1.0" encoding="UTF-8"?>
<model-definition xmlns="http://www.cosminexus.com/mngsvr/schema/ModelDefinition-2.5">
  <web-system>
    <name>MyWebSystem</name>
    <!-- 負荷分散機の定義 -->
    <load-balancer>
      :
    </load-balancer>
    <!-- 物理ティア (combined-tier) の定義 -->
    <tier>
      <tier-type>combined-tier</tier-type>

    <!-- Generate web-server configuration by cmx_trans_param -->
    <configuration>
      <logical-server-type>web-server</logical-server-type>
      <param>
        <param-name>ThreadsPerChild</param-name>
        <param-value>40</param-value>
      </param>
      <param>
        <param-name>MaxClients</param-name>
        <param-value>40</param-value>
      </param>
      <param>
        <param-name>StartServers</param-name>
        <param-value>40</param-value>
      </param>
      <param>
        <param-name>HWSKeepStartServers</param-name>
        <param-value>On</param-value>
      </param>
      <param>
        <param-name>KeepAliveTimeout</param-name>
        <param-value>3</param-value>
      </param>
    </configuration>

    <!-- Generate j2ee-server configuration by cmx_trans_param -->
    <configuration>
      <logical-server-type>j2ee-server</logical-server-type>
      <param>
        <param-name>webserver.connector.ajp13.backlog</param-name>
```

```
        <param-value>40</param-value>
    </param>
    <param>
        <param-name>webserver.connector.ajp13.max_threads</
param-name>
        <param-value>20</param-value>
    </param>
    :
</configuration>
</tier>
<!-- サービスユニットの定義 -->
<unit>
    :
</unit>
</web-system>
<!-- ホストの定義 -->
<host>
    :
</host>
</model-definition>
```

注 太字の部分が自動展開された定義です。

(3) Connector 属性ファイルのパラメタの値

Connector 属性ファイルの、MaxPoolSize および MinPoolSize のパラメタについては、抽象パラメタの値を基に設定値を決定できます。これらのパラメタは、cmx_trans_param コマンドで自動的に設定されません。手動で Connector 属性ファイルに設定し、リソースアダプタをデプロイしてください。

(a) MaxPoolSize

設定値

リクエスト最大同時処理数に設定した値を設定します。

実パラメタの説明

コネクションプールの最大値を指定します。

設定先のファイル：Connector 属性ファイル

(b) MinPoolSize

設定値

リクエスト最大同時処理数に設定した値を設定します。

実パラメタの説明

コネクションプールの最小値を指定します。

設定先のファイル：Connector 属性ファイル

付録 J このマニュアルの参考情報

このマニュアルを読むに当たっての参考情報を示します。

付録 J.1 関連マニュアル

アプリケーションサーバのマニュアルについて次に示します。

- Cosminexus アプリケーションサーバ V8 概説 (3020-3-U01)
アプリケーションサーバの概要について説明しています。
- Cosminexus アプリケーションサーバ V8 ファーストステップガイド (3020-3-U02)
Application Server または Developer を使用して、サンプルプログラムを動かすためのシステムを構築する手順について説明しています。
- Cosminexus アプリケーションサーバ V8 システム設計ガイド (3020-3-U03)
システム設計時に、システムの目的に応じたシステム構成や運用方法を検討するための指針について説明しています。また、チューニングの方法についても説明しています。
- Cosminexus アプリケーションサーバ V8 システム構築・運用ガイド (3020-3-U04)
セットアップウィザードおよび Smart Composer 機能を使用したシステムの構築・運用の手順について説明しています。
- Cosminexus アプリケーションサーバ V8 機能解説 基本・開発編 (Web コンテナ) (3020-3-U05)
アプリケーションサーバで提供する Web コンテナの機能、および Web コンテナに関連する機能 (Web サーバ、サーレット / JSP など) について説明しています。
- Cosminexus アプリケーションサーバ V8 機能解説 基本・開発編 (EJB コンテナ) (3020-3-U06)
アプリケーションサーバで提供する EJB コンテナの機能、および EJB コンテナに関連する機能 (EJB, EJB クライアントなど) について説明しています。
- Cosminexus アプリケーションサーバ V8 機能解説 基本・開発編 (コンテナ共通機能) (3020-3-U07)
Web コンテナおよび EJB コンテナで共通して利用する機能について説明しています。
- Cosminexus アプリケーションサーバ V8 機能解説 拡張編 (3020-3-U08)
アプリケーションサーバで提供する拡張機能 (セッションフェイルオーバー機能、バッチサーバ、CTM など) について説明しています。
- Cosminexus アプリケーションサーバ V8 機能解説 運用 / 監視 / 連携編 (3020-3-U09)
アプリケーションサーバで提供する運用・監視機能、およびほかのプログラムとの連携について説明しています。
- Cosminexus アプリケーションサーバ V8 機能解説 保守 / 移行 / 互換編 (3020-3-U10)
アプリケーションサーバで構築したシステムの保守に関する機能、移行情報、および互換用機能について説明しています。
- Cosminexus アプリケーションサーバ V8 アプリケーション設定操作ガイド

(3020-3-U12)

アプリケーションサーバで動作するアプリケーションの操作方法について説明しています。

- Cosminexus アプリケーションサーバ V8 運用管理ポータル操作ガイド (3020-3-U13)
運用管理ポータルの使用方法について説明しています。

- Cosminexus アプリケーションサーバ V8 リファレンス 定義編 (サーバ定義)

(3020-3-U15)

アプリケーションサーバを構築・運用するとき、またはアプリケーションを開発するときに使用するファイルのうち、J2EE サーバや Management Server などのサーバの定義に使用するファイルの形式について説明しています。

- Cosminexus アプリケーションサーバ V8 リファレンス 定義編 (アプリケーション / リソース定義) (3020-3-U16)

アプリケーションサーバを構築・運用するとき、またはアプリケーションを開発するときに使用するファイルのうち、アプリケーションやリソースの属性設定に使用するファイルの形式について説明しています。

- Cosminexus アプリケーションサーバ V8 仮想化システム構築・運用ガイド

(3020-3-U18)

アプリケーションサーバを仮想化したサーバ上に構築する場合の設計、構築、運用の手順について説明しています。

- Cosminexus アプリケーションサーバ V8 アプリケーション開発ガイド (3020-3-U25)
アプリケーションサーバで動作させるアプリケーションの開発方法について説明しています。

- Cosminexus アプリケーションサーバ V8 リファレンス API 編 (3020-3-U26)

アプリケーションを開発するときに、使用する API の形式について説明しています。

- Cosminexus アプリケーションサーバ V8 メッセージ 1 KDAL-KDCG および Hitachi Web Server 編 (3020-3-U41)

アプリケーションサーバで出力される KDAL から KDCG までのメッセージ、および Hitachi Web Server のメッセージについて説明しています。

- Cosminexus アプリケーションサーバ V8 メッセージ 2 KDJE-KDJW 編 (3020-3-U42)
アプリケーションサーバで出力される KDJE から KDJW までのメッセージについて説明しています。

- Cosminexus アプリケーションサーバ V8 メッセージ 3 KECX-KEDT /

KEOS02000-29999 / KEUC-KFRM 編 (3020-3-U43)

アプリケーションサーバで出力される KECX から KEDT までのメッセージ、KEOS02000 から KEOS29999 までのメッセージ、および KEUC から KFRM までのメッセージについて説明しています。

- Cosminexus アプリケーションサーバ V8 メッセージ 4 監査ログ編 (3020-3-U44)

アプリケーションサーバで出力される監査ログメッセージについて説明しています。

また、このマニュアルと関連するこのほかのマニュアルを次に示します。必要に応じてお読みください。

- Hitachi Web Server (3020-3-U17)
- VisiBroker Version 5 Borland(R) Enterprise Server VisiBroker(R) デベロッパーズガイド (3020-3-U28)
- JP1 Version 8 JP1/Cm2/Extensible SNMP Agent (3020-3-L04)
- JP1 Version 9 JP1/Cm2/Extensible SNMP Agent (3020-3-T04)

なお、このマニュアルでは、次のマニュアルについて、対象 OS およびバージョン番号を省略して表記しています。マニュアルの正式名称とこのマニュアルでの表記を次の表に示します。

正式名称	このマニュアルでの表記
Cosminexus アプリケーションサーバ V8 概説	Cosminexus アプリケーションサーバ 概説
Cosminexus アプリケーションサーバ V8 システム設計ガイド	Cosminexus アプリケーションサーバ システム設計ガイド
Cosminexus アプリケーションサーバ V8 システム構築・運用ガイド	Cosminexus アプリケーションサーバ システム構築・運用ガイド
Cosminexus アプリケーションサーバ V8 機能解説 基本・開発編 (Web コンテナ)	Cosminexus アプリケーションサーバ 機能解説 基本・開発編 (Web コンテナ)
Cosminexus アプリケーションサーバ V8 機能解説 基本・開発編 (EJB コンテナ)	Cosminexus アプリケーションサーバ 機能解説 基本・開発編 (EJB コンテナ)
Cosminexus アプリケーションサーバ V8 機能解説 基本・開発編 (コンテナ共通機能)	Cosminexus アプリケーションサーバ 機能解説 基本・開発編 (コンテナ共通機能)
Cosminexus アプリケーションサーバ V8 機能解説 拡張編	Cosminexus アプリケーションサーバ 機能解説 拡張編
Cosminexus アプリケーションサーバ V8 機能解説 運用 / 監視 / 連携編	Cosminexus アプリケーションサーバ 機能解説 運用 / 監視 / 連携編
Cosminexus アプリケーションサーバ V8 機能解説 保守 / 移行 / 互換編	Cosminexus アプリケーションサーバ 機能解説 保守 / 移行 / 互換編
Cosminexus アプリケーションサーバ V8 アプリケーション設定操作ガイド	Cosminexus アプリケーションサーバ アプリケーション設定操作ガイド
Cosminexus アプリケーションサーバ V8 運用管理ポータル操作ガイド	Cosminexus アプリケーションサーバ 運用管理ポータル操作ガイド
Cosminexus アプリケーションサーバ V8 リファレンス 定義編 (サーバ定義)	Cosminexus アプリケーションサーバ リファレンス 定義編 (サーバ定義)

正式名称	このマニュアルでの表記
Cosminexus アプリケーションサーバ V8 リファレンス 定義編 (アプリケーション / リソース定義)	Cosminexus アプリケーションサーバ リファレンス 定義編 (アプリケーション / リソース定義)
Cosminexus アプリケーションサーバ V8 仮想化システム構築・運用ガイド	Cosminexus アプリケーションサーバ 仮想化システム構築・運用ガイド
Cosminexus アプリケーションサーバ V8 アプリケーション開発ガイド	Cosminexus アプリケーションサーバ アプリケーション開発ガイド
JP1 Version 8 JP1/Cm2/Extensible SNMP Agent	JP1/Cm2/Extensible SNMP Agent
JP1 Version 9 JP1/Cm2/Extensible SNMP Agent	
VisiBroker Version 5 Borland(R) Enterprise Server VisiBroker(R) デベロッパーズガイド	Borland(R) Enterprise Server VisiBroker(R) デベロッパーズガイド

付録 J.2 このマニュアルでの表記

このマニュアルで使用している表記と、対応する製品名を次に示します。

表記		製品名
Application Server	Application Server Enterprise	uCosminexus Application Server Enterprise
	Application Server Standard	uCosminexus Application Server Standard
Developer	Developer Professional	uCosminexus Developer Professional
	Developer Standard	uCosminexus Developer Standard
HiRDB または HiRDB サーバ	HiRDB/Parallel Server	HiRDB/Parallel Server Version 7
		HiRDB/Parallel Server Version 8
	HiRDB/Single Server	HiRDB/Single Server Version 7
		HiRDB/Single Server Version 8
HiRDB Run Time または HiRDB クライアント	HiRDB/Run Time Version 7	
	HiRDB/Run Time Version 8	
IPF	Itanium(R) Processor Family	
JP1/Cm2	JP1/Cm2/ESA	JP1/Cm2/Extensible SNMP Agent
JP1/IM	JP1/IM - CM	JP1/Integrated Management - Central Information Master
	JP1/IM - Manager	JP1/Integrated Management - Manager
	JP1/IM - View	JP1/Integrated Management - View
JP1/SC/DPM	JP1/ServerConductor/Deployment Manager Standard Edition	

表記		製品名	
Oracle	Oracle10g	Oracle 10g	
		Oracle 10g R2	
	Oracle11g	Oracle Database 11g	
		Oracle Database 11g R2	
	Oracle9i	Oracle9i	
		Oracle9i R2	
UNIX	AIX	AIX 5L V5.3	
		AIX V6.1	
	HP-UX または HP-UX (IPF)	HP-UX 11i V2 (IPF)	
		HP-UX 11i V3 (IPF)	
	Linux	Linux (IPF)	Red Hat Enterprise Linux(R) AS 4 (IPF)
			Red Hat Enterprise Linux(R) 5 (Intel Itanium)
			Red Hat Enterprise Linux(R) 5 Advanced Platform (Intel Itanium)
		Linux (x86 / AMD64 & Intel EM64T)	Red Hat Enterprise Linux(R) AS 4 (x86)
			Red Hat Enterprise Linux(R) 5 Advanced Platform (x86)
			Red Hat Enterprise Linux(R) ES 4 (x86)
			Red Hat Enterprise Linux(R) 5 (x86)
			Red Hat Enterprise Linux(R) AS 4 (AMD64 & Intel EM64T)
			Red Hat Enterprise Linux(R) 5 Advanced Platform (AMD/Intel 64)
			Red Hat Enterprise Linux(R) ES 4 (AMD64 & Intel EM64T)
	Red Hat Enterprise Linux(R) 5 (AMD/Intel 64)		
	Solaris	Solaris 10 (SPARC)	
		Solaris 10 (x64)	
		Solaris 9 (SPARC)	
	VMware	VMware vCenter Server	VMware(R) vCenter Server
	Web Redirector		uCosminexus Web Redirector
XDM/RD E2		VOS3 XDM/RD E2	

なお，Application Server および Developer を総称して，アプリケーションサーバと表記します。

このマニュアルで使用している表記と，対応するアプリケーションサーバの機能名を次

に示します。

表記	アプリケーションサーバの機能名
CJMSP ブローカー	Cosminexus JMS プロバイダのブローカー機能
CJMSP リソースアダプタ	Cosminexus JMS プロバイダのリソースアダプタ
Cosminexus Developer's Kit for Java	Cosminexus Developer's Kit for Java TM
CTM	Cosminexus Component Transaction Monitor
Management Server	Cosminexus Management Server
PRF	Cosminexus Performance Tracer
TPBroker	Cosminexus TPBroker

このマニュアルで使用している表記と、対応する Java 関連用語を次に示します。

表記	Java 関連用語
BOM	Byte Order Mark
Connector 1.0	J2EE TM Connector Architecture 1.0
Connector 1.5	J2EE TM Connector Architecture 1.5
EAR	Enterprise ARchive
EJB または Enterprise JavaBeans	Enterprise JavaBeans TM
J2EE または Java 2 Platform, Enterprise Edition	Java TM 2 Platform, Enterprise Edition
J2SE	Java TM 2 Platform, Standard Edition
JAAS	Java TM Authentication and Authorization Service
JAR	Java TM Archive
Java	Java TM
Java 2 Runtime Environment, Standard Edition	Java TM 2 Runtime Environment, Standard Edition
Java 2 SDK, Standard Edition	Java TM 2 Software Development Kit, Standard Edition
JavaBeans	JavaBeans TM
Java EE または Java Platform, Enterprise Edition	Java TM Platform, Enterprise Edition
JavaMail	JavaMail TM
JavaVM	Java TM Virtual Machine
JAXP	Java TM API for XML Processing
JCA	J2EE TM Connector Architecture
JCE	Java TM Cryptography Extension

表記	Java 関連用語
JDBC	JDBC™
	Java™ Database Connectivity
JDK	Java™ Development Kit
JMS	Java™ Message Service
JNDI	Java Naming and Directory Interface™
JNI	Java™ Native Interface
JSF	JavaServer™ Faces Reference Implementation (RI) Version: 1.1_01 FCS
JSP	JavaServer Pages™
JSTL	JavaServer Pages™ Standard Tag Library
JTA	Java™ Transaction API
JTS	Java™ Transaction Service
Servlet またはサーブレット	Java™ Servlet
SMAP	Source Map
WAR	Web Archive

付録 J.3 英略語

このマニュアルで使用している英略語を次に示します。

英略語	英字での表記
API	<u>A</u> pplication <u>P</u> rogramming <u>I</u> nterface
ASCII	<u>A</u> merican <u>S</u> tandard <u>C</u> ode for <u>I</u> nformation <u>I</u> nterchange
BMP	<u>B</u> ean- <u>M</u> anaged <u>P</u> ersistence
BMT	<u>B</u> ean- <u>M</u> anaged <u>T</u> ransaction
CA	<u>C</u> ertification <u>A</u> uthority
CMP	<u>C</u> ontainer- <u>M</u> anaged <u>P</u> ersistence
CMR	<u>C</u> ontainer- <u>M</u> anaged <u>R</u> elationship
CMT	<u>C</u> ontainer- <u>M</u> anaged <u>T</u> ransaction
CORBA	<u>C</u> ommon <u>O</u> bject <u>R</u> equest <u>B</u> roker <u>A</u> rchitecture
CPU	<u>C</u> entral <u>P</u> rocessing <u>U</u> nit
CR	<u>C</u> arriage <u>R</u> eturn
CRL	<u>C</u> ertificate <u>R</u> evocation <u>L</u> ist
CSR	<u>C</u> ertificate <u>S</u> igning <u>R</u> equest

英略語	英字での表記
CSV	<u>C</u> omma <u>S</u> eparated <u>V</u> alue
CUI	<u>C</u> haracter <u>U</u> ser <u>I</u> nterface
DB	<u>D</u> ata <u>b</u> ase
DBMS	<u>D</u> ata <u>b</u> ase <u>M</u> anagement <u>S</u> ystem
DD	<u>D</u> evelopment <u>D</u> escriptor
DIT	<u>D</u> irectory <u>I</u> nformation <u>T</u> ree
DN	<u>D</u> istinguished <u>N</u> ame
DNS	<u>D</u> omain <u>N</u> ame <u>S</u> ystem
DoS	<u>D</u> enial of <u>S</u> ervice attack
DTD	<u>D</u> ocument <u>T</u> ype <u>D</u> efinition
EIS	<u>E</u> nterprise <u>I</u> nformation <u>S</u> ystem
EJB QL	<u>E</u> JB <u>Q</u> uery <u>L</u> anguage
EUC	<u>E</u> xtended <u>U</u> NIX <u>C</u> ode
FF	<u>F</u> orm <u>F</u> eed
GC	<u>G</u> arbage <u>C</u> ollection
GUI	<u>G</u> raphical <u>U</u> ser <u>I</u> nterface
HTML	<u>H</u> yper <u>T</u> ext <u>M</u> arkup <u>L</u> anguage
HTTP	<u>H</u> yper <u>T</u> ext <u>T</u> ransfer <u>P</u> rotocol
IDE	<u>I</u> ntegrated <u>D</u> evelopment <u>E</u> nvironment
IOP	<u>I</u> nternet <u>I</u> nter- <u>O</u> rb <u>P</u> rotocol
ISAPI	<u>I</u> nternet <u>S</u> erver <u>A</u> pplication <u>P</u> rogramming <u>I</u> nterface
ISO	<u>I</u> nternational <u>O</u> rganization for <u>S</u> tandardization
JAR	<u>J</u> ava <u>A</u> rchive
JDBC	<u>J</u> ava <u>D</u> ata <u>b</u> ase <u>C</u> onnectivity
JIS	<u>J</u> apanese <u>I</u> ndustrial <u>S</u> tandards
JMS	<u>J</u> ava <u>M</u> essage <u>S</u> ervice
JNI	<u>J</u> ava <u>N</u> ative <u>I</u> nterface
JVM	<u>J</u> ava <u>V</u> irtual <u>M</u> achine
LAN	<u>L</u> ocal <u>A</u> rea <u>N</u> etwork
LDAP	<u>L</u> ightweight <u>D</u> irectory <u>A</u> ccess <u>P</u> rotocol
LDIF	<u>L</u> LDAP <u>D</u> ata <u>I</u> nterchange <u>F</u> ormat
LF	<u>L</u> ine <u>F</u> eed
MIB	<u>M</u> anagement <u>I</u> nformation <u>B</u> ase

英略語	英字での表記
OID	<u>O</u> bject <u>I</u> dentifier
OMG	<u>O</u> bject <u>M</u> anagement <u>G</u> roup
ORB	<u>O</u> bject <u>R</u> equest <u>B</u> roker
OS	<u>O</u> perating <u>S</u> ystem
OTS	<u>O</u> bject <u>T</u> ransaction <u>S</u> ervice
RDB	<u>R</u> elational <u>D</u> atabase
RMI	<u>R</u> emote <u>M</u> ethod <u>I</u> nvocation
RPC	<u>R</u> emote <u>P</u> rocedure <u>C</u> all
SFO	<u>S</u> ession <u>F</u> ail <u>O</u> ver
SHA	<u>S</u> ecure <u>H</u> ash <u>A</u> lgorithm
SNMP	<u>S</u> imple <u>N</u> etwork <u>M</u> anagement <u>P</u> rotocol
SOA	<u>S</u> ervice <u>O</u> riented <u>A</u> rchitecture
SPI	<u>S</u> ervice <u>P</u> rovider <u>I</u> nterface
SPP	<u>S</u> ervice <u>P</u> roviding <u>P</u> rogram
SSL	<u>S</u> ecure <u>S</u> ockets <u>L</u> ayer
TCS	<u>T</u> ransaction <u>C</u> ontext <u>S</u> erver
UDDI	<u>U</u> niversal <u>D</u> escription, <u>D</u> iscovery and <u>I</u> ntegration
UML	<u>U</u> nified <u>M</u> odeling <u>L</u> anguage
UNC	<u>U</u> niversal <u>N</u> aming <u>C</u> onvention
URI	<u>U</u> niform <u>R</u> esource <u>I</u> dentifier
URL	<u>U</u> niform Resource <u>L</u> ocator
UTC	<u>U</u> niversal <u>T</u> ime <u>C</u> oordinated
UTF	<u>U</u> CS <u>T</u> ransformation <u>F</u> ormat
WAR	<u>W</u> eb <u>A</u> rchive
XML	<u>E</u> xtensible <u>M</u> arkup <u>L</u> anguage

付録 J.4 常用漢字以外の漢字の使用について

このマニュアルでは、常用漢字を使用することを基本としていますが、次に示す用語については、常用漢字以外の漢字を使用しています。

鍵（かぎ） 個所（かしょ） 汎用（はんよう） 必須（ひつす）

付録 J.5 KB（キロバイト）などの単位表記について

1KB（キロバイト）、1MB（メガバイト）、1GB（ギガバイト）、1TB（テラバイト）はそ

それぞれ $1,024$ バイト, $1,024^2$ バイト, $1,024^3$ バイト, $1,024^4$ バイトです。

索引

A

adminagentcheck 307
adminagentctl 309
auditsetup 477

C

car_tar_Z 445
cjaddapp 44
cjaddsec 154
cjchmodapp 47
cjdellog 493
cjcdumpap 496
cjclearpool 92
cjclosecn 95
cjclstartap 497
cjcopyres 98
cjdeleteapp 49
cjdeletejb 100
cjdeletelibjar 51
cjdeleteres 102
cjdeletesec 155
cjdeployrar 104
cjdumpsv 19
cjdumpweb 523
cjenvsetup 20
cjenvupdate 21
cjexecjob 183
cjexportapp 53
cjexportrar 105
cjgencmpsql 56
cjgetappprop 58
cjgetjbprop 107
cjgetrarprop 109
cjgetresprop 111
cjgetstubsjar 61
cjgetsysinfo 23
cjimportapp 63
cjimportjb 114
cjimportlibjar 67

cjimportres 117
cjjsp2java 509
cjjspc 503
cjkilljob 187
cjlistapp 69
cjlistjb 120
cjlistjob 189
cjlistlibjar 73
cjlistpool 121
cjlistrar 125
cjlistres 129
cjlistsec 157
cjlistthread 159
cjlisttrn 161
cjlisttrnfile 164
cjmapsec 167
cjmsbroker 226
CJMSP プロローカー管理 223
CJMSP プロローカーの起動 226
cjrarupdate 25
cjreloadapp 75
cjrenameapp 77
cjreplaceapp 79
cjresetsv 168
cjresumepool 133
cjsetappprop 82
cjsetjbprop 135
cjsetrarprop 136
cjsetresprop 138
cjsetup 31, 179
cjsleep 33
cjstartapp 85
cjstartjb 141
cjstartrar 142
cjstartrecover 34
cjstartsv 36, 181
cjstartweb 524
cjstopapp 87
cjstopjb 144
cjstoprar 145
cjstopsv 38

- cjstopthread 170
- cjstopweb 525
- cjsuspendpool 146
- cjtestres 149
- cjtracesync 171
- cjundeployrar 151
- cjunmapsec 172
- cjwebchangedbid 528
- cjwebclearpool 529
- cjwebclosepool 531
- cjwebdeletepool 532
- cjwebeditpool 533
- cjweblistpool 535
- cjwebsetup 540
- CMP2.x Entity Bean 用 SQL 文の生成 56
- cmx_add_serverref 364
- cmx_admin_passwd 365
- cmx_build_model 366
- cmx_build_system 367
- cmx_change_model 371
- cmx_define_application 372
- cmx_define_resource 374
- cmx_delete_serverref 375
- cmx_delete_system 376
- cmx_deploy_application 379
- cmx_deploy_resource 380
- cmx_export_model 382
- cmx_list_model 384
- cmx_list_status 389
- cmx_register_application 392
- cmx_register_resource 393
- cmx_resume_lb 394
- cmx_scaleout_host 395
- cmx_start_application 397
- cmx_start_resource 398
- cmx_start_target 400
- cmx_stop_application 403
- cmx_stop_resource 404
- cmx_stop_target 406
- cmx_test_lb 409
- cmx_trans_param 410
- cmx_undefine_application 411
- cmx_undefine_resource 412
- cmx_undeploy_application 413
- cmx_undeploy_resource 415
- cmx_unregister_application 416
- cmx_unregister_resource 417
- Component Container 管理者のセットアップ 20
- Connector 属性ファイルのパラメタの値 624
- convpw 481
- CORBA ネーミングサービスの起動 562, 582
- CORBA ネーミングサービスの停止 595, 574
- core アーカイブ機能 445
- core ファイルの取得 450
- cosmienv 515
- Cosminexus DABroker Library のセットアップ 545
- Cosminexus JMS プロバイダで使用するコマンド 221
- Cosminexus JMS プロバイダで使用するコマンドの一覧 223
- Cosminexus JMS プロバイダで使用するコマンドの概要 222
- Cosminexus JMS プロバイダで使用するコマンドの注意事項 249
- Cosminexus アダプタコマンドのセットアップとアンセットアップ 314
- Cosminexus モニタ起動コマンドのセットアップとアンセットアップ 315
- cprfed 199
- cprfflush 204
- cprfgetpid 205
- cprflevel 207
- cprfstart 213
- cprfstop 219
- ctmchpara 254
- ctmdminfo 256
- ctmdmstart 258
- ctmdmstop 262
- ctmgetpid 263
- ctmholdque 264
- ctmlogcat 268
- ctmlsque 269

- ctmrasget 273
 - ctmregltd 274
 - ctmridinfo 276
 - ctmrlesque 277
 - ctmstart 279
 - ctmstop 288
 - ctmstsed 289
 - ctmstsflush 298
 - ctmstsstart 299
 - ctmstsstop 301
 - ctmtscgwd 302
 - ctmver 303
 - CTM デーモンの開始 279
 - CTM デーモンの起動 564, 584
 - CTM デーモンの終了 288
 - CTM デーモンの停止 573, 594
 - CTM で使用するコマンド 251
 - CTM ドメイン情報の出力 276
 - CTM ドメイン情報の表示と削除 256
 - CTM ドメインマネージャの開始 258
 - CTM ドメインマネージャの起動 582, 562
 - CTM ドメインマネージャの終了 262
 - CTM ドメインマネージャの停止 574, 596
 - CTM レギュレータの開始 274
- D**
-
- dabsetup 545
 - daemon 309
- E**
-
- eheapprof 447
 - Explicit ヒープ詳細情報付き拡張スレッドダンプの出力 447
- H**
-
- HWSKeepStartServers 621
- J**
-
- J2EE アプリケーション 590
 - J2EE アプリケーションと Web システムの関連づけ 372
 - J2EE アプリケーションと Web システムの関連づけ解除 411
 - J2EE アプリケーションのアンデプロイ 413
 - J2EE アプリケーションのインポート 63
 - J2EE アプリケーションのエクスポート 53
 - J2EE アプリケーションの開始 85, 397, 568, 589
 - J2EE アプリケーションの削除 49, 416
 - J2EE アプリケーションの停止 87, 403, 571, 592
 - J2EE アプリケーションのデプロイ 379
 - J2EE アプリケーションの登録 392
 - J2EE サーバで使用するコマンド 15
 - J2EE サーバの開始 36
 - J2EE サーバの起動 565, 585
 - J2EE サーバの起動間隔 33
 - J2EE サーバのコネクションクローズまたはコネクションの状態表示 95
 - J2EE サーバのスレッドダンプの取得 19
 - J2EE サーバのセットアップとアンセットアップ 31
 - J2EE サーバの停止 38, 572, 593
 - J2EE サーバのトランザクション回復 34
 - JavaBeans リソースの一覧表示 120
 - JavaBeans リソースのインポート 114
 - JavaBeans リソースの開始 141
 - JavaBeans リソースの削除 100
 - JavaBeans リソースの属性設定 135
 - JavaBeans リソースの属性の取得 107
 - JavaBeans リソースの停止 144
 - javacore 450
 - javagc 453
 - javatrace 458
 - JavaVM で使用するコマンド 443
 - JavaVM で使用するコマンドの一覧 444
 - JavaVM で使用するコマンドの詳細 445
 - Java アプリケーションで使用するコマンド 491
 - Java アプリケーションで使用するコマンドの一覧 492
 - Java アプリケーションで使用するコマンドの詳細 493
 - Java アプリケーションの開始 497

Java アプリケーションのシステムログ出力
先サブディレクトリの削除 493
Java アプリケーションのスレッドダンプの
取得 496
jheapprof 461
jheapprofanalyzer 467
JSP ソースのチェック 509
JSP の事前コンパイル 503

K

KeepAliveTimeout 621
kill 310

M

Management Server 管理ファイルの回復
330
Management Server 管理ファイルの退避
329
Management Server で使用するコマンド
305
Management Server の移行コマンド 319
Management Server の運用管理コマンド
321
Management Server の管理ユーザアカウン
トの設定 365
Management Server の起動 / 停止 / セット
アップ 315
MaxClients 620
MaxPoolSize 624
MinPoolSize 624
mngautorun 311
mngsvr_adapter_setup 314
mngsvr_monitor_setup 315
mngsvrctl 315
mngsvrupdate 319
mngsvrutil 321
mstrexport 329
mstrimport 330

O

OS 状態情報の取得 23
OTM ゲートウェイの開始 302

P

PRF デーモンの開始 213
PRF デーモンの起動 581, 561
PRF デーモンの停止 219, 597
PRF デーモンプロセス ID の取得 205
PRF トレース取得レベルの指定例
(cprflevel コマンド) 209
PRF トレース取得レベルの指定例
(cprfstart コマンド) 214, 216
PRF トレース取得レベルの表示と変更 207

R

RAR 属性設定 136
RAR ファイルの属性の取得 109

S

SFO サーバの起動 564, 584
SFO サーバの停止 573, 594
Smart Composer 機能で使用するコマンド
353
Smart Composer 機能で使用するコマンドの
詳細 362
Smart Composer 機能で使用できる抽象パラ
メタ 618
snapshotlog 473
snapshot ログの収集 473
ssoexport 482
ssogenkey 484
ssoimport 485
StartServers 621

T

ThreadsPerChild 620

U

uachpw 488

V

vmx_create_unit 424
vmx_delete_unit 425

vmx_deploy_unit 426
 vmx_export_rule 428
 vmx_get_machine_info 428
 vmx_list_unit_version 429
 vmx_list_virtualserver_status 431
 vmx_rolling_update 434
 vmx_scalein_unit 434
 vmx_scaleout_unit 435
 vmx_set_machine_info 436
 vmx_set_unit_currentversion 437
 vmx_start_unit 438
 vmx_stop_unit 439
 vmx_update_unit 440

W

webserver.connector.ajp13.backlog 622
 webserver.connector.ajp13.max_threads 622
 Web アプリケーションで使用するコマンド 501
 Web アプリケーションで使用するコマンドの一覧 502
 Web アプリケーションで使用するコマンドの詳細 503
 Web コンテナサーバで使用するコマンド 521
 Web コンテナサーバの開始 524
 Webコンテナサーバのスレッドダンプの取得 523
 Web コンテナサーバのセットアップとアンセットアップ 540
 Web コンテナサーバの停止 525
 Web サーバの起動 569, 590
 Web サーバの停止 571, 592
 Web システムの構築 367
 Web システムの削除 376
 Web システムの情報モデルの取得 384
 Web システムの情報モデルの生成 366
 Web システムの情報モデルの変更 371, 395
 Webシステムまたはサービスユニットの開始 400
 Webシステムまたはサービスユニットの停止 406

あ

アプリケーションサーバの起動の流れ (UNIX の場合) 577, 612
 アプリケーションサーバの起動の流れ (Windows の場合) 557
 アプリケーションの RMI-IIOP スタブおよびインタフェースの取得 61
 アプリケーションの一覧表示 69
 アプリケーションの入れ替え 79
 アプリケーションの属性設定 82
 アプリケーションの属性の取得 58
 アプリケーションのモード切り替え 47
 アプリケーションのリロード 75
 アプリケーション名の変更 77
 暗号鍵ファイルの作成 484

い

インプロセス HTTP サーバ用トレースファイルの同期 171

う

運用管理エージェントの稼働確認 307
 運用管理エージェントの起動と停止 309

か

ガーベージコレクションの強制発生 453
 仮想化システムで使用するコマンド 419
 仮想化システムで使用するコマンドの詳細 423
 仮想サーバの稼働状況の一覧表示 431
 仮想サーバのスケールアウト 435
 仮想サーバのスケールイン 434
 稼働中の J2EE サーバのトランザクション情報の表示 161
 稼働統計情報の取得開始 299
 稼働統計情報の取得終了 301
 稼働統計情報の編集と出力 289
 簡易構築定義ファイルに展開される実パラメータ 620
 簡易構築定義ファイルの出力 382
 監査ログの設定で使用するコマンド 475

監査ログの設定で使用するコマンドの一覧 476
 監査ログの設定で使用するコマンドの詳細 477
 管理対象マシン情報の取得 428
 管理対象マシン情報の設定 436
 管理ユニット運用ルールの参照 428
 管理ユニットの一括起動 438
 管理ユニットの一括停止 439
 管理ユニットのカレントバージョンの設定 437
 管理ユニットの削除 425
 管理ユニットの作成 424
 管理ユニットのデプロイ 426
 管理ユニットのバージョン情報一覧表示 429
 管理ユニットの変更 440
 管理ユニットのローリングアップデート 434

き

旧バージョンとの互換用のコマンドの一覧 356

く

クライアント側共通定義ファイル
 (mngsvrutilcl.properties) 326
 クライアント側定義ファイル
 (.mngsvrutilrc) 325

こ

構築済み実行環境の切り替えで使用するコマンド 513
 構築済み実行環境の切り替えで使用するコマンドの一覧 514
 構築済み実行環境の切り替えで使用するコマンドの詳細 515
 コネクションプール内のコネクション削除 92
 コネクションプールの一覧表示 121
 コマンドを同時に複数実行した場合の排他制御 359

さ

サーバ側定義ファイル
 (mngsvrutil.properties) 326
 サーバ管理コマンドの排他強制解除 168
 サービスの設定 311
 サービスユニット状況の表示 389
 作業ディレクトリおよびユーザ定義ファイルの移行 21
 サブコマンド 324
 サブコマンド change に指定できる引数 346
 サブコマンド check に指定できる引数 347
 サブコマンド collect に指定できる引数 345
 サブコマンド dump に指定できる引数 349
 サブコマンド get に指定できる引数 335
 サブコマンド hold に指定できる引数 342
 サブコマンド list に指定できる引数 332
 サブコマンド release に指定できる引数 344
 サブコマンド reload に指定できる引数 342
 サブコマンド set に指定できる引数 337
 サブコマンド start に指定できる引数 339
 サブコマンド stop に指定できる引数 340

し

システム起動時の環境変数の設定 561, 581
 システムの起動手順 556, 576, 606, 612
 システムの起動の流れ (Windows の場合) 606
 システムの起動方法 561, 581, 607
 システムの停止手順 569, 590, 609, 615
 システムの停止の流れ 569, 590, 609, 615
 システムの停止方法 571, 592, 610, 616
 実行形式ファイルおよびライブラリのバージョン情報の出力 267, 303
 詳細レベルのトレース情報を取得する場合の指定値 (cprflevel コマンド) 208
 詳細レベルのトレース情報を取得する場合の指定値 (cprfstart コマンド) 215
 詳細レベルまたは保守レベルのトレース情報を取得する場合の指定値 (cprflevel コマンド) 209

詳細レベルまたは保守レベルのトレース情報を取得する場合の指定値 (cprfststart コマンド) 215
 シングルサインオン情報リポジトリの参照 482
 シングルサインオン情報リポジトリの登録 485

す

スケジュールキュー情報の出力 269
 スケジュールキューの同時実行数の変更 254
 スケジュールキューの閉塞 264
 スケジュールキューの閉塞解除 277
 スマートエージェントの起動 562, 582
 スマートエージェントの停止 574, 596
 スレッド情報の表示 159
 スレッドの削除 170

せ

性能解析トレース情報の編集出力 199
 性能解析トレースで使用するコマンド 197
 セッションフェイルオーバーサーバの起動の流れ (UNIX の場合) 580
 セッションフェイルオーバーサーバの起動の流れ (Windows の場合) 560

ち

抽象パラメタによって展開される実パラメタ 619
 抽象パラメタの展開 410, 618
 抽象パラメタの展開例 622

て

停止中の J2EE サーバのトランザクション情報の表示 164
 データベース ID の変更 528
 デーモンプロセスの起動 309

と

統合ネーミングスケジューラサーバの起動の流れ (UNIX の場合) 579

統合ネーミングスケジューラサーバの起動の流れ (Windows の場合) 559
 統合ユーザ管理で使用するコマンド 479
 統合ユーザ管理で使用するコマンドの一覧 480
 統合ユーザ管理で使用するコマンドの詳細 481
 トラブルシュート情報の取得と出力 273
 トレース情報の収集 458

は

パスワードの暗号化 481
 パスワードの変更 488
 バッチアプリケーション情報の一覧表示 189
 バッチアプリケーションで使用するコマンド 183
 バッチアプリケーションの実行 183, 608
 バッチアプリケーションの停止 610
 バッチアプリケーションの停止またはバッチサーバの強制停止 187
 バッチサーバで使用するコマンド 175
 バッチサーバで使用するコマンドの概要 176
 バッチサーバで使用するリソース操作コマンド 194
 バッチサーバの開始 181
 バッチサーバの起動 608
 バッチサーバのセットアップとアンセットアップ 179
 バッチサーバの停止 610
 バッチサーバを操作するコマンド 178
 バッファ内容の強制ファイル出力 204, 298

ひ

日立クラス別統計情報解析ファイルの CSV 出力 467
 日立クラス別統計情報付き拡張スレッドダンプの出力 461

ふ

プール管理情報の一覧表示 535
 プール管理情報の削除 532
 プール管理情報の登録と変更 533

プール内の接続の強制切断 531
プール内の接続のクリア 529
負荷分散機との接続テスト 409
負荷分散機の設定再開始 394
プロセス ID の取得 263
プロセスの強制停止 310

め

メッセージの編集と出力 268
メンバコネクションプールの一時停止 146
メンバコネクションプールの再開始 133

ゆ

ユーザとロールのアンマッピング 172
ユーザとロールの一覧表示 157
ユーザとロールの削除 155
ユーザとロールの追加 154
ユーザとロールのマッピング 167

ら

ライブラリ JAR の一覧表示 73
ライブラリ JAR のインポート 67
ライブラリ JAR の削除 51

り

リソースアダプタのアンデプロイ 151
リソースアダプタの一覧表示 125
リソースアダプタのエクスポート 105
リソースアダプタの開始 142
リソースアダプタの停止 145
リソースアダプタのデプロイ 104
リソースアダプタのバージョンアップ 25
リソースと Web システムの関連づけ 374
リソースと Web システムの関連づけ解除
412
リソースのアンデプロイ 415
リソースの一覧表示 129
リソースのインポート 117
リソースの開始 398, 568, 589
リソースのコピー 98
リソースの削除 102, 417

リソースの接続テスト 149
リソースの属性設定 138
リソースの属性の取得 111
リソースの追加 44
リソースの停止 404, 572, 593
リソースのデプロイ 380
リソースの登録 393

ろ

ログの運用で使用するコマンド 471
ログの運用で使用するコマンドの一覧 472
ログの運用で使用するコマンドの詳細 473
論理ユーザサーバの削除 375
論理ユーザサーバの追加 364

ソフトウェアマニュアルのサービス ご案内

1. マニュアル情報ホームページ

ソフトウェアマニュアルの情報をインターネットで公開しています。

URL <http://www.hitachi.co.jp/soft/manual/>

ホームページのメニューは次のとおりです。

マニュアル一覧	日立コンピュータ製品マニュアルを製品カテゴリ、マニュアル名称、資料番号のいずれかから検索できます。
CD-ROMマニュアル	日立ソフトウェアマニュアルと製品群別CD-ROMマニュアルの仕様について記載しています。
マニュアルのご購入	マニュアルご購入時のお申し込み方法を記載しています。
オンラインマニュアル	一部製品のマニュアルをインターネットで公開しています。
サポートサービス	ソフトウェアサポートサービスお客様向けページでのマニュアル公開サービスを記載しています。
ご意見・お問い合わせ	マニュアルに関するご意見、ご要望をお寄せください。

2. インターネットでのマニュアル公開

2種類のマニュアル公開サービスを実施しています。

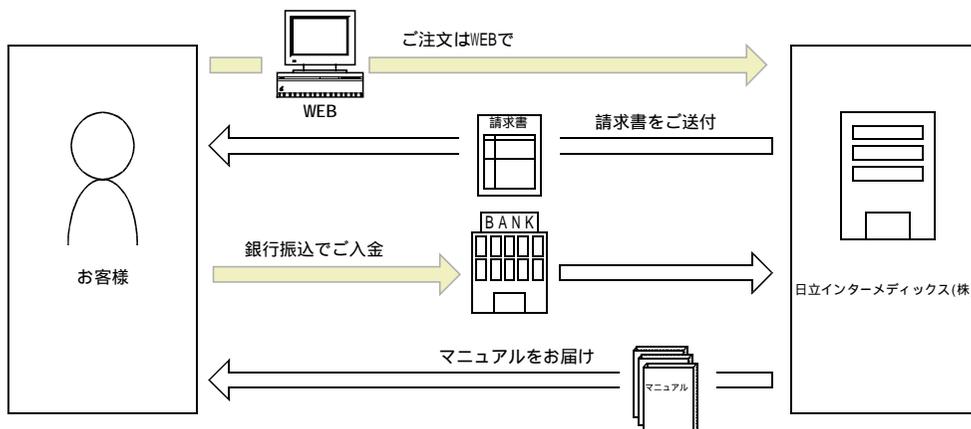
(1) マニュアル情報ホームページ「オンラインマニュアル」での公開

製品をよりご理解いただくためのご参考として、一部製品のマニュアルを公開しています。

(2) ソフトウェアサポートサービスお客様向けページでのマニュアル公開

ソフトウェアサポートサービスご契約のお客様向けにマニュアルを公開しています。公開しているマニュアルの一覧、本サービスの対象となる契約の種別などはマニュアル情報ホームページの「サポートサービス」をご参照ください。

3. マニュアルのご注文



マニュアル情報ホームページの「マニュアルのご購入」にアクセスし、お申し込み方法をご確認のうえWEBからご注文ください。ご注文先は日立インターメディアックス(株)となります。

ご注文いただいたマニュアルについて請求書をお送りします。

請求書の金額を指定銀行へ振り込んでください。

入金確認後7日以内にお届けします。在庫切れの場合は、納期を別途ご案内いたします。